

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV INTELIGENTNÍCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF INTELLIGENT SYSTEMS

MNOHA-KONTEXTOVÝ REPUTAČNÍ SYSTÉM V PROSTŘEDÍ WWW

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE

AUTHOR

Bc. Petr Kadlec

BRNO 2011



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV INTELIGENTNÍCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF INTELLIGENT SYSTEMS

MNOHA-KONTEXTOVÝ REPUTAČNÍ SYSTÉM V PROSTŘEDÍ WWW

MULTI-CONTEXT REPUTATION SYSTEM FOR WWW

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE

AUTHOR

Bc. Petr Kadlec

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. Jan Samek

BRNO 2011

Abstrakt

Tato diplomová práce seznamuje čtenáře se základními pojmy z oblasti reputace a důvěry, problematiky reputačních systémů a mnoha-kontextových modelů reputace. Dále popisuje vybrané praktické více-kontextové modely v prostředí WWW, na jejichž základě je navržen vlastní více-kontextový model použitý v realizované aplikaci.

Abstract

This Diploma Thesis introduces the basic concepts of reputation and trust, reputation systems and problems of multi-contextual models of reputation. It also describes practical aspects of multi-contextual models in web environment. Described contextual models are then used for design and implementation of real web application.

Klíčová slova

Reputace, důvěra, reputační systémy, více-kontextový, Spring, karma.

Keywords

Reputation, trust, reputation systems, multi-context, Spring, karma.

Citace

Kadlec Petr: Mnoha-kontextový reputační systém v prostředí WWW, diplomová práce, Brno, FIT VUT v Brně, 2011

Mnoha-kontextový reputační systém v prostředí WWW

Prohlášení

Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně pod vedením Ing. Jana Samka. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....
Petr Kadlec

Poděkování

Na tomto místě bych chtěl poděkovat svému vedoucímu diplomové práce, panu Ing. Janu Samkovi, za odborné vedení a pomoc při zpracovávání této práce.

© Petr Kadlec, 2011

Tato práce vznikla jako školní dílo na Vysokém učení technickém v Brně, Fakultě informačních technologií. Práce je chráněna autorským zákonem a její užití bez udělení oprávnění autorem je nezákonné, s výjimkou zákonem definovaných případů.

Obsah

Obsah.....	1
1 Úvod.....	3
2 Reputační systémy	5
2.1 Reputace a důvěra.....	6
2.1.1 Základní reputační modely	7
2.1.2 Mnoha-kontextové modely důvěry a reputace.....	9
2.1.3 Principy stárnutí příspěvků	12
2.1.4 Problém rozdílného počtu hodnocení	13
3 Známé mnoha-kontextové modely v prostředí WWW	15
3.1 Mnoha-kontextové modely v praxi.....	15
3.1.1 Best Buy.....	15
3.1.2 Nike.....	17
3.1.3 Fajnšmekr	18
3.1.4 Crest.....	19
3.1.5 Restaurantica.....	19
3.1.6 Nettravel	20
3.1.7 DealExtreme	21
3.1.8 TESTFREAKS	22
3.1.9 Hotel Thailand	23
3.1.10 What car?	24
3.1.11 Web Hosting Geeks	25
3.1.12 BOOKING.....	25
3.2 Porovnání mnoha-kontextových modelů	26
4 Návrh reputačního modelu a aplikace.....	28
4.1 Neformální specifikace.....	28
4.2 Návrh vlastního mnoha-kontextového modelu reputace	29
4.3 Návrh výpočtu celkového skóre objektu	30
4.3.1 Stanovení střední hodnoty q-karmy	30
4.3.2 Q-karma uživatele.....	31
4.3.3 Q-karma recenze	32
4.3.4 P-karma recenze.....	33
4.3.5 Hodnocení recenze.....	35
4.3.6 Celkové skóre objektu	35
4.4 Návrh aplikace	36

4.4.1	Diagram případu použití	36
4.4.2	Návrh databázového schématu	37
4.5	Použité technologie.....	40
4.5.1	Spring.....	40
4.5.2	Spring Security	41
4.5.3	Hibernate.....	41
4.5.4	Oracle.....	41
4.5.5	Tiles	41
4.5.6	JSP + JSTL	41
4.5.7	JQuery.....	42
5	Implementace aplikace.....	43
5.1	Architektura aplikace.....	43
5.2	Autorizace a autentizace	44
5.3	Více-kontextové vyhledávání a řazení.....	46
5.4	Databázová dotazovací kritéria.....	47
6	Navržená vylepšení modelu a jejich implementace	48
6.1	Rozdělení uživatelské q-karmy podle kategorií.....	48
6.2	Celkové skóre zohledňující množství recenzí	49
7	Případová studie.....	51
7.1	Objekty, uživatelé a recenze	51
7.2	Vliv Q-karmy uživatele na skóre	51
7.3	Vliv P-karmy recenze na skóre	53
7.4	Vliv Q-karmy recenze na skóre	54
7.5	Srovnání vlastního reputačního modelu s existujícími	55
8	Možnosti rozšíření modelu.....	56
8.1	Rekalkule hodnot recenzí při změně váhy kontextů	56
8.2	Vazby mezi kategoriemi	56
8.3	Uchovávání vlastního váhového rozložení	57
8.4	Pokročilejší implementace modelu stárnutí	58
9	Závěr	60

1 Úvod

V dnešní době má téměř každý člověk zkušenosti s nákupem zboží po internetu. Člověk může nakupovat z pohodlí domova, většinou za nižší cenu a ještě nemusí nikde čekat ve frontě. Ztratil tak ale poslední možnost si danou věc na živo prohlédnout a vyzkoušet, a pro jistotu se o ní ještě pobavit s odborným prodejcem. Mnohdy se jednalo o nákup doporučené a ověřené věci od známých a přátel, nebo si danou věc nejprve člověk vyzkoušel v kamenném obchodě a pak si ji na internetu koupil levněji. Určitě se už Vám ale také stalo, že jste si chtěli koupit věc, o které jste nevěděli vůbec nic a vyzkoušet jste si ji nemohli. Je to opravdu „Televize s nejlepším obrazem současnosti“, jak o ní píše výrobce ve specifikaci na stránkách prodejce. A vůbec, je dokonce věrohodný ten prodejce, že doručí zboží, když požaduje platbu na účet předem? Koneckonců nemusí jít ani o žádný nákup zboží. Třeba jste se jen chtěli podívat na internetu na recenzi filmu, aby jste na něj pak mohli vyrazit do kina. Jelikož ale recenzent nebyl zrovna fanouškem žánru a film nedocenil, řídili jste se jeho hodnocením a zůstali raději doma.

Dříve než byl internet masově dostupný, byli jste odkázáni na Vaše zkušenosti či zkušenosti Vašeho okolí komu věřit a komu ne. Případně jste dali na rady přítomného prodejce. Internetové služby však operují na daleko větším prostoru a navíc ještě poskytují virtuálně anonymní komunikaci. Zde se již nemůžete spoléhat na pravou identitu druhé strany, neboť na internetu může mít každý identitu kolik je libo. Zmrazení účtu škodlivého uživatele na bazarovém fóru nijak nezabrání, aby se daný uživatel nezaregistroval do systému pod novým jménem a dál nepobíral platby za neexistující zboží. Možností, jakým vnést do chaosu alespoň nějaký řád, jsou právě *Reputační systémy*. Škodlivému uživateli nezabráníte v opětovném přístupu ani díky nim, ale vůči nováčkům s nízkou reputací jsou ostatní členové alespoň daleko více obezřetnější.

Reputační systémy sbírají, rozdělují a slučují reakce od uživatelů. Pomáhají tak lidem v rozhodování co vybrat a komu důvěřovat. Oddělují tak kvalitní věci od nekvalitních a spolehlivé uživatele od nečestných a nezkušených.

The Internet Movie Database [11], největší filmová databáze obsahující databázi všech filmů, seriálů, animací z počítačových her a jiných filmových materiálů. Přestože tento server nemá žádné své placené recenzenty, obsahují všechny filmová díla celkové hodnocení agregované z hodnocení jednotlivých uživatelů. Každý zaregistrovaný uživatel má právo hodnotit filmové dílo na stupnici 1 až 10. Čím víc bodů danému dílu uživatele dávají, tím víc se dílo pohybuje po stupnici kvality směrem nahoru.

Jiným serverem, který by bez reputačního systému ani nemohl existovat, je největší online aukce na světě eBay [10]. Tato aukce neposkytuje žádnou záruku za vlastní aukce, pouze slouží jako zprostředkovatel, kde sami kupující a prodávající přebírají všechna rizika spojená s transakcí. Aby takový systém mohl fungovat, má každý kupující a prodávající příležitost se po každé dokončené

transakci navzájem ohodnotit kladným, záporným či neutrálním bodem a případně ještě přidat komentář. Skóre vypočítané z hodnocení jednotlivých transakcí je pak nedílnou součástí uživatelského profilu, kde každá strana transakce má možnost do tohoto hodnocení nahlédnout a rozhodnout se, zda je pro něj proti strana dostatečně důvěryhodná a obchod uskutečnit.

Co když by ale případný kupující na tomto serveru spěchal na doručení zásilky. Vyplývá ze souhrnného skóre sčítající palce, ve kterém má prodávající jenom samá kladná hodnocení, že prodejce skutečně odešle zboží nejpozději do 3 dnů od přijetí peněz, tak jak je to napsáno v popisu aukce? Či vyplývá z jeho skóre, že s Vámi bude aktivně komunikovat a ochotně vyhoví Vaším požadavkům, v případě že budete potřebovat na poslední chvíli změnit dodací adresu? Řešení tohoto problému se nabízí v rozdělení či doplnění hlavního hodnocení o sadu předem nadefinovaných vedlejších aspektů, které jdou taktéž hodnotit a pro prodávajícího můžou mít důležitý význam. V případě serveru *eBay* jsou těmito doplňujícími kontexty - *shoda produktu s jeho popisem, komunikace, čas potřebný k odeslání a poplatky spojené s odesláním*.

Ukázkou velice vhodného využití mnoha-kontextového hodnocení v reputaci je v oblasti cestování. Na dovolené bude každého člena rodiny zajisté zajímat něco odlišného. Zatímco třeba děti bude spíše zajímat pláž, zábava a sportovní vyžití, pohodlnější rodiče nejspíše dají přednost kvalitnějšímu stravování či vybavení pokoje. Serverem, kde se s takovým reputačním modelem můžeme setkat, je například *NetTravel* [15].

Přestože na internetu již existuje celá řada serverů využívajících mnoha-kontextové reputace, jen málo který umí plně využít potenciálu, které toto rozdělení nabízí. Jedna z těch méně využitých oblastí je problematika řazení a vyhledávání podle kontextů objektů. Tato diplomová práce si tak všímá tohoto volného prostoru, a proto představuje návrh a implementaci alternativního pohledu na možnosti paralelního řazení objektů podle kontextů.

Kapitola 2 se zabývá teoretickým popisem pojmů z oblastí reputačních systémů. Jsou zde zmíněny základní reputační modely, mnoha-kontextové modely důvěry a principy řešení problémů stárnutí příspěvku a rozdílného počtu hodnocení. Zároveň je v ní definována terminologie důležitá pro orientaci v reputačních modelech.

Kapitola 3 zkoumá reálně používané mnoha-kontextové modely reputace v prostředí webu, aby na základě shrnutí v závěru této kapitoly byl navržen v kapitole 4 vlastní mnoha-kontextový model. V kapitole 4 je také zmíněn návrh budoucí aplikace.

Kapitola 5 se zabývá samotnou implementací aplikace. Všímá si přitom pouze těch zajímavějších oblastí týkajících se možností prostředí, v jakém je aplikace vyvíjena, či ne zcela běžných ovládacích prvků.

V kapitole 6 jsou na základě uživatelského testování popsány rozšíření implementovaného mnoha-kontextového modelu navrženého v kapitole 4. Kapitola 7 se věnuje experimentování s mnoha-kontextovým modelem. A v kapitole 8 jsou popsány další možná rozšíření modelu a aplikace, jejich návrh a způsob implementace.

2 Reputační systémy

V této kapitole budou vymezeny některé důležité pojmy z oblasti reputačních systémů. Před samotným popisem ale bude nejprve nutné sjednotit základní terminologii, jež se bude používat nejen v této kapitole, ale v průběhu celé diplomové práce a je důležitá pro správnou orientaci v problematice reputačních systémů, reputace i aplikace samotné. Pokud není uvedeno jinak, mají dané termíny přesně následující význam:

Kontexty

- Kontext - důležitá vlastnost objektů, která stojí za zvážení, zda li by se nemohla hodnotit zvlášť.

Hodnocení

- Recenze - slovní a bodové hodnocení objektu (přesněji vysvětleno dále v této kapitole pod pojmem Reviews) .
- Hodnota kontextu - bodová reprezentace kontextu (typické je zobrazení například pomocí počtu jedné až pěti hvězd).
- Hodnocení - výsledek recenze vypočítaný z dílčích hodnot kontextů kategorie (v případě, že mají kontexty váhu, pak i společně s jejich váhami).
- Hlavní hodnocení - hlavní hodnotící kontext objektu, z něhož se jako z jediného počítá hodnocení. V reputačních systémech bez možnosti hodnocení kontextů je hlavní hodnocení zároveň i jediné.
- Palec - neboli hodnocení recenze, je reakce uživatele na recenzi v podobě kladného či záporného hlasu.
- Palcování - ve významu hodnotit recenzi palcem.
- Celkové skóre (skóre) - celkové hodnocení objektu (například pro umístění v žebříčku), vypočítané na základě vyhodnocovací funkce (vysvětlena níže).
- Vyhodnocovací funkce - funkce počítající z dílčích hodnocení celkové skóre.

Karma

- P-karma (Participation karma) - může být objektu či uživatele objektu (přesněji vysvětleno dále v této kapitole).
- Q-karma (Quality karma) - může být objektu či uživatele objektu (přesněji vysvětleno dále v této kapitole).
- Celková karma - součet všech karem majících vliv na recenzi.

2.1 Reputace a důvěra

Reputace, důvěra nebo zkušenost jsou pojmy, které vychází z běžného života. Přestože použití těchto pojmů ve světě informačních technologií není zcela běžné, díky různým reputačním systémům ale tyto termíny nabývají na významu.

V současné literatuře lze nalézt několik definic pojmů *důvěry* a *reputace*. Původní význam slova důvěra ale pochází ze sociologie, kde je definováno takto:

Důvěra je typ postoje a zároveň mezilidského vztahu, který vyvolává pocit jistoty plynoucí z přesvědčení, že partner komunikace (osoba, instituce) splní určitá očekávání [5].

V odborných pracích s tematikou informačních technologií pak nacházíme řadu dalších definic, které z originálního významu s patřičnou úpravou vychází. Často citovaným zdrojem je také definice od autora Lik Mui:

Důvěra je subjektivní víra v entitu, že se bude chovat za určitých okolností určitým, předem očekávaným způsobem, na základě jeho předchozího chování [6].

Třetí zajímavou definicí důvěry citované od autorů Vosse, Brücknera a Schlossera [7] je:

Důvěra je míra ochoty přistoupit k akci (rozhodnutí), která vystavuje strany (entity) riziku újmy. Je založena na posouzení možných rizik, odměn a reputací spojené se všemi stranami zapojenými v dané situaci

Pojem **reputace** pak popisuje Lik Mui [6] pomocí pojmu reciprocita (vzájemná výměna laskavostí či odplat) a to tak, že reputace je hodnotou důvěry vypočítané na základě akcí dané entity A a postřehů ostatních entit v rámci jistého prostředí. Reputace tak zjevně ovlivňuje míru důvěry ostatních v danou entitu. Hlavním rozdílem mezi důvěrou a reputací je, že důvěra je posuzována z lokálního hlediska, zatímco reputace má globální význam.

Autoři Vosse, Brückner a Schlossera [7] pak ve své práci popisují účel **reputace** jako shromažďování a zpracovávání informací o dosavadním chování dané entity, tak jak ji vnímají ostatní entity.

Účelem **reputačního systému** je pak agregovat hodnocení, které shromáždil za účelem vypočtení celkového skóre pro každou zúčastněnou entitu [7].

Základní jednotkou všech reputačních systémů je tzv. *reputační tvrzení*. Reputační tvrzení můžeme mít dvojího druhu:

- Explicitní: někdo udělá prohlášení o kvalitě něčeho („Volva jsou spolehlivá auta“, řekl Karel).
- Implicitní: někdo udělá něco, z čehož si vyvodíme názor (Karel si koupil Volvo).

Všechny reputační tvrzení pak mohou být zobrazeny do modelu na obrázku 1.



Obrázek 1: Zobrazení modelu pro popis reputačních tvrzení.

2.1.1 Základní reputační modely

Výčet níže popsanych reputačních modelů od autora Farmera [1] tvoří základní stavební kameny velkých reputačních systémů. Jelikož k většině z těchto pojmů neexistuje přesný a jednoznačný český překlad, byly u všech modelů ponechány jejich originální názvy.

2.1.1.1 Favorites and Flags

Tento model má za účel identifikovat v kolekci objektů extrémní hodnoty, ať už na straně kladné či záporné. Obvykle má podobu přidání hlasu pro hodnocený subjekt, často však i umožňuje daný obsah určitým způsobem označit, či na tento obsah upozornit přitelem pomocí odkazu. Hlavním ukazatelem je zde čítač, jež počítá, kolikrát bylo k danému obsahu pomocí výše zmíněných způsobů přistoupeno. K Favorites and Flags tak například patří dále zmíněné modely:

- *Vote to Promote* - v tomto modelu uživatel hlasuje pro jistý objekt z určité množiny podobných objektů. Objekty s více hlasy se tak posunují v žebříčku výše a stávají se viditelnějšími. Na rozdíl od modelu *This-or-That Voting* (viz níže) *Vote to Promote* vybírá z množiny objektů potenciálně neomezené.

- *Favorites* – počítá, kolikrát uživatel označil daný obsah.
- *Report Abuse* – slouží k označování nevhodného obsahu. V případě překročení jisté meze opakovaným hlášením, dojde k cenzuře obsahu či k jeho smazání.

2.1.1.2 This-or-That Voting

Pomocí tohoto modelu se dává uživatelům možnost hlasovat například o užitečnosti či přesnosti jistého objektu. Na rozdíl od *Vote to Promote* umožňuje vybrat z ohraničeného množství předdefinovaných možností.

2.1.1.3 Ratings

Umožňuje uživatelům vyjádřit kvalitu o určitém objektu, kde hodnocení kvality je rozděleno na jisté skalární hodnoty, například pěti hvězd či deseti-bodovou stupnici. Celkové hodnocení objektu je pak dáno průměrnou hodnotou jednotlivých hodnocení.

2.1.1.4 Reviews

Reviews je obecně několik spojených Ratings dohromady, umožňujících hodnotit objekt z více kontextů. Jednotlivé kontexty samostatně by jinak nedávaly moc smysl. Součástí Reviews bývá také z pravidla komentář, odůvodňující či doplňující hodnocení.

2.1.1.5 Points

Specifický model sloužící pro sčítání aktivit uživatele na serveru. Systém počítá například počty stažení, přidané komentáře, zodpovězené otázky, přidané příspěvky a jiné.

2.1.1.6 Karma

Slouží k určení reputace mezi uživateli. Obvykle doplňuje ostatní modely za účelem motivace uživatelů. V nejobecnější formě existují 2 druhy kare: karma měřící zapojení uživatele do dění na serveru a karma měřící kvalitu příspěvků. V případě kombinace obou kare pak získávají největší skóre uživatele píšící pravidelně kvalitní příspěvky.

Typy kare:

- *Participation karma* – sčítá body za akce uživatele, kde každá akce má obvykle jiné bodové ohodnocení. Typické užití je například bodování za psaní příspěvků. Model *Participation karmy* je zpravidla implementován jako bodový systém. Tento model má i negativní variantu, jež se od kladné liší počítáním prohrašek namísto aktivity.
- *Quality karma* – se zabývá výhradně kvalitou příspěvků od uživatele. Na rozdíl od *Participation karmy* zde množství příspěvků nehraje žádnou roli, pokud příspěvek není

nějakým způsobem ohodnocen. *Quality karma* uživatele tak obvykle vzniká jako vedlejší efekt hodnocení uživatelských příspěvků jinými uživateli.

- *Robust karma* – Spojuje vlastnosti *Participation* a *Quality karmy* do jedné. Část spojená s kvalitou nutí uživatele věnovat čas pro napsání kvalitního příspěvku a participation část jej nutí přispívat často. *Robust karma* ale není obvykle uživatelům zobrazována a slouží zpravidla k vnitřnímu hodnocení, označování či může ovlivňovat výsledky při vyhledávání.

2.1.2 Mnoha-kontextové modely důvěry a reputace

Kontextovost je podle [4] schopnost nebo možnost nahlížet na jednotlivce či objekty z různých úhlů pohledů. Tato vlastnost vychází z požadavků na zvýšení granularity důvěry v reálném světě. Na základě kontextovosti pak jsme schopni důvěřovat objektu v různých aspektech. Tyto aspekty pak označujeme jako kontexty.

Pro lepší názornost mnoha-kontextovosti si můžeme popsat příklad týkající se grafických karet. U grafických adaptérů jsou důležité pro většinu lidí aspekty jako: *cena*, *výkon*, *hlučnost* a *spotřeba*. Dostáváme tedy celkem 4 kontexty, kde každý kontext můžeme ohodnotit různým stupněm důvěry, čímž zároveň zvýšíme granularitu hodnocení. Daný člověk si pak může na základě svých preferencí vybrat, zda vyžaduje bezpodmínečně nejvýkonnější kartu nebo chce, aby byl grafický adaptér co nejtišší, a nebo je pro něj rozhodující cena. Někdy je vhodné vytvářet ze základních kontextů i kontexty složené. U grafických karet by takovým aspektem byl například poměr *cena* / *výkon*.

Pokud bychom nezavedli hodnocení samostatně pro každý kontext, museli bychom hodnotit objekt jako celek, čímž bychom do značné míry ztratily možnost posuzovat a hodnotit objekt z více pohledů.

V následujících podkapitolách budou některé významné práce, které se zabývají popisem a formalizací více-kontextových modelů důvěry či reputace. Nicméně, existuje i mnoho dalších prací, které se o více-kontextovosti důvěry zabývají [8, 9].

2.1.2.1 Alfarez Abdul-Rahman, Stephen Hailes

V práci [4] popsali autoři Rahman a Hailes model důvěry pro agentní systémy, který respektuje kontextovou povahu důvěry. Formální definice či struktura kontextu není v této práci explicitně definována a je ponecháno na konkrétní implementaci, používající tento model, jak bude stanovena. Tento model definuje způsob vyhodnocování přímé důvěry a důvěry v doporučovateli vždy z pohledu agenta x , přičemž používá množinu čtyř diskrétních hodnot pro stanovování stupně důvěry. Dále bude formálně popsán princip modelu včetně způsobu vyhodnocení důvěry. Veškeré formalismy jsou převzaty z článku [4].

Popis modelu

Model definuje zápis přímé důvěry jako důvěru agenta x v jiného agenta a v určitém kontextu c s jistým stupněm td důvěry (pozn. td - zkratka z anglického „direct trust“) následovně:

$$t(a, c, td) \quad (1)$$

kde $td \in \{vt, t, u, vu\}$ označuje sémantiku stupňů důvěry podle tabulky 1.

Stupeň důvěry	Význam
vt	„Velmi důvěryhodný“
t	„Důvěryhodný“
u	„Nedůvěryhodný“
vu	„Velmi nedůvěryhodný“

Tabulka 1: Stupně důvěry a jejich význam.

Agent x pak také může věřit jinému agentovi b s jistým stupněm důvěry rtd vzhledem k jeho schopnosti poskytovat doporučení o jiných agentech v rámci daného kontextu c :

$$rt(b, c, rtd) \quad (2)$$

kde hodnota rtd reprezentuje „sémantický rozdíl“ mezi doporučením agenta b a vlastním přesvědčením agenta a o věrohodnosti doporučujícího agenta. Doporučení totiž nemusí nutně reprezentovat přesvědčení doporučujícího agenta, protože doporučovatel může lhát nebo poskytovat protichůdné doporučení různým agentům.

Pro potřeby vyhodnocení přímé důvěry a důvěry v doporučovatele definují autoři několik datových struktur: množina agentů A , množina kontextů C , množina možných výsledků zkušeností E , množina přímých zkušeností Q a množina pro doporučující agenty R . Jednotlivé množiny jsou definovány následovně:

Množina A. Množina agentů komunikujících s agentem x .

$$A = \{a_1, \dots, a_n\} \quad (3)$$

Množina C. Množina kontextů známých vzhledem k agentu x .

$$C = \{c_1, \dots, c_n\} \quad (4)$$

Množina E. Tato množina definuje, jakých hodnot mohou nabývat výsledky přímých zkušeností či zkušeností z doporučení a může nabývat hodnot: *velmi dobré* (*vg*), *dobré* (*g*), *špatné* (*b*), a *velmi špatné* (*vb*). Tyto hodnoty korespondují s hodnotami v tabulce 1.

$$E = \{vg, g, b, vb\} \quad (5)$$

Množina Q. Tato množina obsahuje pro každého agenta a a kontext c definovanou čtveřici

$s = (s_{vg}, s_g, s_b, s_{vb})$, kde s_j je akumulátor počtu hodnocení pro zkušenost e , přičemž platí $j = e$ a zároveň $e \in E$. Potom pro $S = \{(s_{vg}, s_g, s_b, s_{vb})\}$ je Q definována takto:

$$Q \subseteq C \times A \times S \quad (6)$$

Množina R. Toto je množina pro důvěryhodné doporučující agenty. Cílem je zpracovat sémantický rozdíl vnímání v poskytnuté doporučení agenta a jeho důvěryhodnosti. Pro lepší pochopení předvedeme na příkladu. Předpokládejme agenta a , který doporučí agentu x agenta b , který je „velmi důvěryhodný“ vzhledem ke kontextu c , přičemž ale agent x má vztah k agentu b vyhodnocen pouze jako „důvěryhodný“. V budoucnu tak agent x může doporučení od agenta a upravovat. V našem případě tak agent x na základě svých zkušeností s agentem b sníží doporučení od agenta a o jeden stupeň na úroveň „důvěryhodný“. Sémantický rozdíl je pak roven hodnotě -1.

Jelikož máme definovány 4 úrovně hodnocení důvěry, množina možných výsledků (sémantických rozdílů) pak odpovídá množině $G = \{-3, -2, -1, 0, 1, 2, 3\}$. Pro každého agenta a a kontext c získáváme 4 množiny upravených výsledků hodnocení zkušeností T_{vg} , T_g , T_b a T_{vb} , kde každá T_e , pro $e \in E$, reprezentuje upravené hodnocení pro každé doporučení e agenta a . Obor hodnot pro T_e je množina G . Potom nechť $T = \{T_{vg}, T_g, T_b, T_{vb}\}$. Na základě předchozích definic pak můžeme definovat množinu R jako:

$$R \subseteq C \times A \times T \quad (7)$$

Vyhodnocení přímé důvěry

K určení přímé důvěry td agenta a vzhledem ke kontextu c použijeme relaci (c, a, s) z množiny Q , kde $s = (s_{vg}, s_g, s_b, s_{vb})$. Hodnota td se pak určí dle indexu e prvku s_e kde s_e je největší prvek v s jako:

$$\exists td \in E \quad \forall s_e \in s, (s_e = \max(s)) \Rightarrow (td = e) \quad (8)$$

Jestliže funkce $\max(s)$ vrátí více než jednu hodnotu (vrátí více stejně ohodnocených hodnot), je td přiřazena hodnota s neurčitostí podle tabulky 2 (symbol "?" znamená „nula nebo jiná hodnota“).

e	td	Znamená
$vg \wedge g \wedge ?$	u^+	Většinou dobré, občas špatné
$vb \wedge b \wedge ?$	u^-	Většinou špatné, občas dobré
Všechny ostatní kombinace	u^0	Stejně špatné jako dobré

Tabulka 2: Stupně důvěry s neurčitostí.

Vyhodnocení důvěry v doporučovatele

Pro zjištění stupně důvěry v doporučovatele rtd pro agenta a vzhledem ke kontextu c , využijeme relace (c, a, t) , pro $a \in R$, kde $t = (T_{vg}, T_g, T_b, T_{vb})$. Hodnota rtd je získána pomocí operace mod^1 nad absolutními hodnotami prvků z množiny $T^a = T_{vg} \cup T_g \cup T_b \cup T_{vb}$. Tato hodnota pak udává vzdálenosti mezi hodnotami doporučení agenta a hodnotami stanovených na základě skutečných zkušeností vycházejících z jeho doporučení.

$$rtd = mod(\{\forall x \in T^a \mid |x|\}) \quad (9)$$

2.1.3 Principy stárnutí příspěvků

Nejen díky technologickému pokroku, ale i vlivem módy či kulturního posunu, je potřeba se vypořádat se stárnutím objektů a tím pádem i se stárnutím jejich hodnocení. Stejně jako výkonový král mezi grafickými adaptéry starý 5 let se nemůže měřit s právě vydanými modely, musí docházet i k aktualizaci srovnávacích měřítek. Různým typům objektů by měl náležet odlišný způsob aktualizace porovnávání. Ne všechny měřítka se budou měnit stejně, protože jinak se bude měnit názor na restauraci a jinak na mobilní telefon. V [1] jsou popsány 4 specifické algoritmy pro přepočty měřítek v průběhu času a v následujících podkapitolách si je podrobně přiblížíme.

2.1.3.1 Souhrnný přepočty

Celkové skóre daného objektu je sníženo o pevnou procentuální část za jednotku času. K rekalkulaci dochází vždy, když objektu přibude nové hodnocení. Algoritmus je vysoce výkonný, ale zřídka hodnoceným objektům zůstávají dlouho vysoké hodnoty. Nutností je tedy přítomnost časovače, který funkci aktualizace celkového skóre sám spouští v pravidelných intervalech.

¹ Mod - z Modus, což je operace pro získání hodnoty s největší relativní četností v dané množině.

2.1.3.2 Dynamický přepočít

Kdykoliv je přidáno nějaké hodnocení, dochází k přepočtu všech hodnocení vztažených ke stejnému objektu. Funkce sice poskytuje plynulejší křivku, ale časová složitost se stala kvadratickou $O(n^2)$.

2.1.3.3 Dynamické okno

Okno s pevným časovým rozsahem (např. poslední 2 roky) či pevným množstevním rozsahem (např. posledních 50 hodnocení). Nová hodnocení se zařadí na začátek fronty a zároveň se zahodí stejný počet hodnocení z konce. Celková reputace je pak vypočítána z hodnocení, která zůstanou, kdy k přepočtu dochází stejně jako v předchozím případě při přidání nového hodnocení. Tato metoda vytváří skóre z těch nejnovějších dílčích hodnocení, nicméně pro zřídka hodnocené objekty může přesto vracet staré výsledky.

2.1.3.4 Časově omezený přepočít

Ačkoliv tato metoda patří z pohledu procesorového času a množství přenesených dat z databáze k těm nejvíce náročným, patří přesto i k těm nejvíce používaným. K výpočtu celkového skóre je použito všech hodnocení v daném časovém rozsahu a výsledek je použit jen pro ten jeden okamžik, kdy je potřeba (typicky např. zobrazení celkového skóre v profilu objektu).

2.1.4 Problém rozdílného počtu hodnocení

V průběhu této podkapitoly není význam slova *hodnocení* omezen pouze na hodnocení recenze, ale může mít význam i přímého hodnocení objektů, kdy například i anonymní uživatelé mohou jednoduše ohodnotit objekt bez nutnosti se kamkoliv registrovat či cokoli vyplňovat.

Při řazení objektů do různých žebříčků dojde určitě na situaci, kdy podobné objekty obdrží diametrálně odlišný počet hodnocení. Protože ale objekt s pár hodnoceními dostal nejvyšší známky, je v žebříčku před objektem s mnoha hodnoceními. Žebříček pak ale není úplně spravedlivý, protože daleko větší vypovídající hodnotu má ten více hodnocený objekt. Prvotní hodnocení bývají často vytvářena amatéry, či svoji roli také hraje počáteční nadšení, a proto musejí být objekty s pár hodnoceními nějakým způsobem lehce znevýhodněny nebo naopak ty s mnoha zvýhodněny.

Příkladem algoritmu řešící tento problém je algoritmus používaný serverem Yahoo! [18] popsáný v [1]. Celkové skóre R' přepočítané vzhledem k počtu hodnocení se vypočítá jako:

$$R' = R - a - l * a \quad (10)$$

kde:

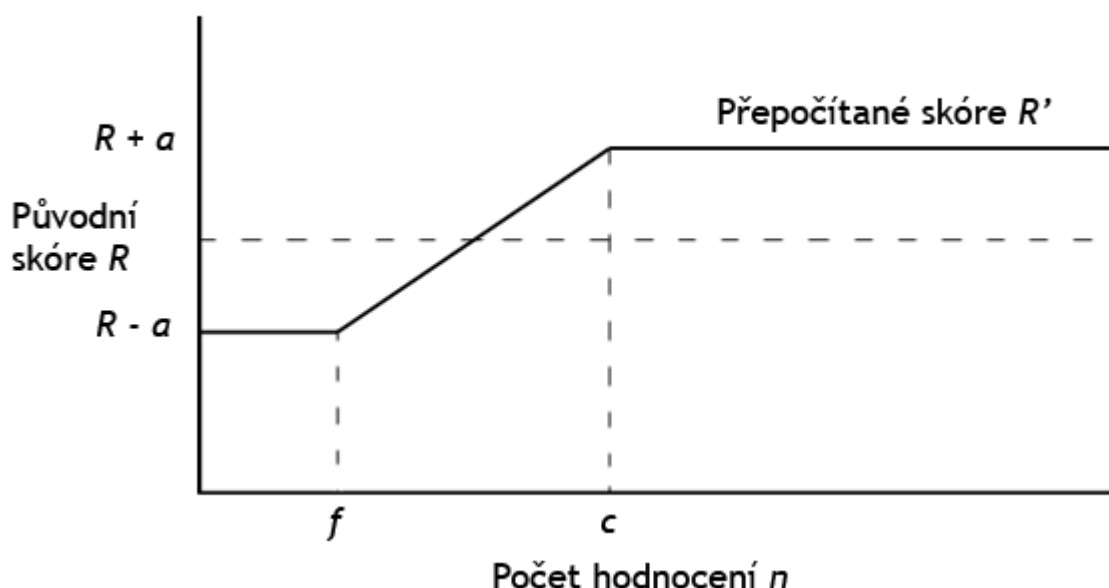
- R je původní hodnota celkového skóre,
- a je zvýhodňovací/znevýhodňovací koeficient,
- l je kompenzační funkce popsána jako:

$$l = \min \left(\max \left(\frac{(n - f)}{c}, 0.00 \right), 1.00 \right) * 2.00 \quad (11)$$

kde:

- n je počet hodnocení,
- f je dolní hranice počtu hodnocení, od které bude celkové skóre zvýhodňováno,
- c je horní hranice počtu hodnocení, od které už počet hodnocení nebude hrát ve zvýhodnění roli,
- funkce \min a \max zajistí, že hodnota zvýhodnění bude v intervalu $< 0, 1 >$.

Výhodou tohoto algoritmu je, že nevyžaduje žádné znalosti předchozího celkového skóre a lze jej jednoduchým způsobem spočítat. Funkce produkuje křivku jako na obrázku 2.



Obrázek 2: Vliv kompenzační funkce na výslednou hodnotu celkového skóre.

Doporučenými hodnotami podle [1] pro a , f a c jsou:

- Kompenzační konstanta $a = 0.10$ – vyšší hodnota bude upřednostňovat více hodnocené objekty. Naopak nižší hodnota se ve výsledku nemusí nijak projevit.
- Konstanta minimálního počtu recenzí f (3-10) – vyšší hodnota minimalizuje zneužití a lépe reprezentuje názor většiny.
- Konstanta maximálního počtu hodnocení $c = (30-60)$ – nepřiměřeně vysoká hodnota činí vysoce hodnocené objekty nepřekonatelnými.

3 Známé mnoha-kontextové modely v prostředí WWW

Tato kapitola obsahuje výčet nalezených a prozkoumaných webových serverů, používajících vlastní mnoha-kontextový model. V závěru kapitoly jsou tyto modely podrobeny srovnání a výsledek je prezentován v souhrnné tabulce.

3.1 Mnoha-kontextové modely v praxi

Vzorce pro výpočet pořadí v žebříčku u jednotlivých serverů byly empiricky odvozeny. Od skutečných algoritmů počítajících celkové skóre se mohou lehce lišit.

3.1.1 Best Buy

Z pohledu mnoha-kontextového hodnocení patří k nejpokročilejším server amerického maloobchodního prodejce spotřební elektroniky Best Buy [12]. Server nabízí širokou paletu výrobků od bílé elektroniky přes spotřební elektroniku až po posilovací stroje či hudební nosiče. Téměř všechny tyto výrobky je pak také možné hodnotit podle důležitých rysů specifických pro danou kategorii zboží. Jednotlivé hodnotící kontexty jsou spravovány administrátory serveru, ale hodnocení je ponecháno na uživatelích.

Aby mohl uživatel hodnotit, musí se nejdříve zaregistrovat. Hodnocení produktu sestává z určení povinného hlavního hodnocení na stupnici jedna až pět hvězd a nepovinného, doplňujícího kontextového hodnocení, jež hodnotí jednotlivé důležité rysy taktéž s možností nula až pět. Nepovinná kontextová hodnocení nejsou do celkového nijak započítávána a pouze hlavní hodnocení upřesňují. Nutností je také doplnit krátkou recenzi s možností vyzdvihnouti pozitiv či negativ.

Přestože server nabízí hodnocení výrobků, neumožňuje je však na základě hodnocení i vyhledat. Vyhledávání podle jednotlivých kontextů výrobku není podporováno (například vyhledání LCD televize s nejlepším obrazem bez ohledu na cenu a možnosti). Průměr celkových hodnocení se pak promítne jen jako grafické znázornění počtu hvězd, výslednou hodnotící známkou a počtem hodnotitelů.

Výrobky	Televize	Ledničky, sporáky	Přehrávače DVD	Reproduktory	Sluchátka	Hudební CD	Posilovací Stroje
Kvalita obrazu	x		x				
Kvalita zvuku	x			x	x		
Možnosti	x	x					x
Výkon/cena		x	x	x	x	x	x
Kvalita		x					x
Výkon		x					
Používání			x	x			
Nastavení			x				
Odolnost				x	x		x
Pohodlí nošení					x		
Kvalita hudby						x	
Kvalita textů						x	
Obal + doplňky						x	

Tabulka 3: Přehled vybraných výrobků a jejich reputačních kontextů na serveru Best Buy.

Počet kontextů	Variabilní (2-4)
Stárnutí příspěvků	Ne
Uživatelská karma	Ne
Množství příspěvků	Ne
Vzorec výpočtu celkového skóre	$y = \frac{1}{m} \sum_{j=0}^m x_j$ <p>kde x_j je hodnota hodnocení j a m je počet hodnocení.</p>
Použité reputační modely	Reviews, This-or-That Voting
Vyhledávání podle kontextů	Ne
Nutnost registrace	Ano
Celkové skóre z kontextů ²	Ne

Tabulka 4: Souhrn vlastností reputačního serveru Best Buy.

² Celkové skóre z kontextů znamená, že recenze nemá celkové hodnocení. Pro potřebu vypočítání celkového skóre objektu jsou použity přímo hodnoty kontextů.

3.1.2 Nike

Sportovní gigant Nike [13] umožňuje na svých stránkách taktéž hodnotit svoje výrobky ve více kontextech. Společným měřítkem všeho zboží je opět celkové hodnocení vyjádřené jedním až pěti body. U vybraného sortimentu ale umožňuje navíc hodnocení upřesnit pomocí specifických atributů. Příkladem takového zboží je sportovní obuv či bundy.

Doplňující kontextové hodnocení je zde prezentováno jako sada sedmibodových posuvníků nastavených na prostřední hodnotu, kde hodnotu kontextu určujeme posuvem na jednu či na druhou stranu. Kontextové hodnocení není do hlavního hodnocení nijak započítáno a opět pouze hlavní hodnocení doplňuje.



Obrázek 3: Ukázka doplňujícího hodnocení u obuvy Nike.

Počet kontextů	Variabilní (3-5)
Stárnutí příspěvků	Ne (není potřeba, protože se jedná o sezónní zboží a nabídka je často kompletně měněna)
Uživatelská karma	Ne
Množství příspěvků	Ano
Vzorec výpočtu celkového skóre y	$y = \frac{1}{m} \sum_{j=0}^m x_j$ <p>kde x_j je hodnota hodnocení j a m je počet hodnocení.</p>
Použité reputační modely	Reviews, This-or-That Voting
Vyhledávání podle kontextů	Ne
Nutnost registrace	Ano
Celkové skóre z kontextů	Ne

Tabulka 5: Souhrn vlastností reputačního serveru Nike.

3.1.3 Fajnšmekr

Fajnšmekr [14] je český reputační server zabývající se hodnocením restaurací, kaváren, pizzerií a dalších gastronomických zařízení. Hodnotí se kontexty – *jídlo*, *pití*, *atmosféra* a *obsluha*. Každý kontext je možné vyjádřit slovním ekvivalentem stupnice jedna až pět, kde jedna značí v případě obsluhy „Pěkně mě naštvá“ a pět je „Skvělá, vynikající“. V případě, že jste si neobjednali, je možné hodnocení jídla či pití vynechat. Součástí hodnocení je také komentář, počet lidí, útrata a datum návštěvy. Každé hodnocení je pak dáno průměrnou hodnotou všech kategorií. Hodnotit restaurace lze jen po registraci s ověřením e-mailu.

Existují 2 skupiny hodnotitelů – normální a certifikovaní. Normálním hodnotitelem se stává uživatel po registraci na webových stránkách. Z normálních uživatelů se stávají certifikovaní po ověření od správců systému. Jedná se o jistou alternativu systému pozvánek, kde jen pozvaní mohou rozdávat pozvánky. Hodnocení certifikovaných uživatelů má v celkovém hodnocení vyšší váhu.

Celkové skóre je vypočítáno jako vážený průměr jednotlivých reputačních záznamů, kde zvýhodňujícím koeficientem jsou vynásobena hodnocení certifikovaných uživatelů. Postupně znevýhodňovány jsou naopak starší příspěvky.

Systém umožňuje nad takto ohodnocenými restauracemi rovněž vyhledávat. Mezi vyhledávací možnostmi patří absolutní vyhledávání podle celkového skóre, nebo podle jednotlivých kontextů a to vždy buď za určité období výhradně, jen od certifikovaných uživatelů či ode všech.

Počet kontextů	Konstantní (4)
Stárnutí příspěvků	Ano (časové stárnutí)
Uživatelská karma	Ano (ovlivňuje hodnocení)
Množství příspěvků	?
Vzorec výpočtu celkového skóre y	$y = \frac{\sum_{j=1}^m \left(\frac{\sum_{i=1}^n w_i x_i}{\sum_{i=1}^n w_i} \right) z_j v_j}{\sum_{j=1}^m z_j v_j}$ <p>kde x značí hodnotu kontextu, w_i je váha kontextu i, v_j je váha uživatele j, z značí váhu založenou na čerstvosti příspěvku, n je počet kontextů a je počet hodnocení.</p>
Použité reputační modely	Reviews, Participation karma
Vyhledávání podle kontextů	Ano
Nutnost registrace	Ano
Celkové skóre z kontextů	Ano

Tabulka 6 : Souhrn vlastností reputačního serveru Fajnšmekr.

3.1.4 Crest

Stejnou tematikou jako *Fajnsmekr* se zabývá i gastronomický server *Crest* [19]. Přestože server k současnému datu obsahuje téměř 30x více zaregistrovaných restaurací a působí na trhu o několik let déle, obsahuje pouze o 20% více uživatelských recenzí. Hodnotit mohou opět jen zaregistrovaní uživatelé. Hodnocení je tvořeno celkem pěti kontexty: *kvalita nápojů*, *kvalita jídla*, *rychlost obsluhy*, *ochota obsluhy*, *atmosféra restaurace*) s pěti stupni hodnocení, kde alespoň jeden musí být vyplněn. Součástí hodnocení je i povinný doplňující komentář. K hlavnímu srovnávání pak slouží šestý kontext -tzv. *celkový dojem*, který je pak vypočítán klasickým průměrem z vyplněných hodnot jednotlivých kontextů. Při srovnávání je do žebříčku promítnuta částka, kterou host v dané restauraci utratil, a datum hodnocení.

Počet kontextů	Konstantní (5)
Stárnutí příspěvků	Ano
Uživatelská karma	Ne
Množství příspěvků	Ne
Vzorec výpočtu celkového skóre y	$y = \frac{1}{m} \sum_{j=0}^m \left(\frac{1}{n} \sum_{i=1}^{n_j} x_{ij} \right)$ <p>kde n je počet kontextů a m je počet hodnocení a x_{ij} je hodnota kontextu i pro hodnocení m.</p>
Použité reputační modely	Reviews
Vyhledávání podle kontextů	Ne (systém pouze umí zobrazit žebříček 10 nejlepších vztažený k jednotlivým kontextům)
Nutnost Registrace	Ano
Celkové skóre z kontextů	Ano

Tabulka 7: Souhrn vlastností reputačního serveru Crest.

3.1.5 Restaurantica

Třetím systémem zabývajícím se hodnocením gastronomických zařízení v Los Angeles je server *Restaurantica* [21]. Nutností pro možnost napsání vlastní recenze je opět registrace. Hodnotit restaurace můžeme celkem ve třech vedlejších aspektech: *jídlo*, *obsluha* a *atmosféra* a jednom hlavním souhrnném. Celkové skóre restaurace je získáno jednoduchým průměrem pouze ze souhrnného hodnocení. Zobrazení restaurací v rámci žebříčku nejlepších restaurací podle jejich

hodnocení je již provedeno sofistikovanějším způsobem. Na umístění v pořadí zde už nehraje roli jen průměrné skóre restaurace, ale i počet hodnocení každé restaurace a jejich stáří.

Počet kontextů	Konstantní (3)
Stárnutí příspěvků	Ano
Uživatelská karma	Ne
Množství příspěvků	Ano
Vzorec výpočtu celkového skóre y	$z = \frac{1}{m} \sum_{j=0}^m x_j$ <p>kde x_j je hodnota hodnocení j a m je počet hodnocení.</p>
Vzorec výpočtu hodnoty y pro umístění v pořadí	Empirickým výzkumem se nepodařilo odhadnout.
Použité reputační modely	Reviews, Report abuse, This-or-Thar Voting (doporučení restaurace)
Vyhledávání podle kontextů	Ne (systém umí pouze zobrazit žebříček vztažené k jednotlivým kontextům)
Nutnost registrace	Ano
Celkové skóre z kontextů	Ne

Tabulka 8: Souhrn vlastností reputačního serveru Restaurantica.

3.1.6 Nettravel

Dalším českým serverem využívajícím mnoha-kontextové reputace je server *NetTravel* [15] zabývající se zprostředkováváním zájezdů. Mnoha-kontextové recenze se zde využívá pro hodnocení hotelů v jednotlivých destinacích. K výběru, jež může uživatel hodnotit, patří 7 předdefinovaných kontextů: *poloha*, *personál*, *vybavení hotelu*, *vybavení pokoje*, *pláž*, *sport a zábava*, *stravování*. Ne všechny kontexty však dávají v rámci určitých zájezdů valný smysl. Ukázkou může být například použití aspektu pláže u zájezdu na hory či atributu sport a zábava u poznávacích zájezdů.

Aby mohl uživatel přidat svou vlastní recenzi hotelu, musí být splněny 2 podmínky. První podmínkou je nezbytné absolvování zvoleného zájezdu, protože jenom pak jej může uživatel objektivně zhodnotit. Druhou podmínkou je registrace do systému, protože bez registrace se na žádný zájezd nemůže přihlásit.

Při hodnocení kontextů dostane uživatel na výběr z jedné až pěti hvězd, či kontext nehodnotit. Výsledné skóre je pak vypočítáno klasickým průměrem z hodnocených kontextů. Obdobně je spočítán i celkový výsledek hotelu ze všech jeho hodnocení.

Počet kontextů	Konstantní (7)
Stárnutí příspěvků	Ne
Uživatelská karma	Ne
Množství příspěvků	Ne
Vzorec výpočtu celkového skóre y	$y = \frac{1}{m} \sum_{j=0}^m \left(\frac{1}{n} \sum_{i=1}^n x_{ij} \right)$ <p>kde n je počet kontextů, m je počet hodnocení a x_{ij} je hodnota kontextu n v hodnocení m.</p>
Použité reputační modely	Reviews
Vyhledávání podle kontextů	Ne
Nutnost registrace	Ano
Celkové skóre z kontextů	Ano

Tabulka 9: Souhrn vlastností reputačního serveru Nettravel.

3.1.7 DealExtreme

DealExtreme [16] je čínský internetový server zabývající se prodejem počítačů, elektroniky a jejich příslušenství. Ke každému předmětu umožňuje server zaregistrovaným uživatelům přidat svoji recenzi nebo alespoň přidat hodnocení. Recenze je tvořena 4mi vedlejšími kontexty: *cena*, *jednoduchost použití*, *výrobní kvalita*, *užitečnost*, jedním hlavním kontextem: *souhrnné skóre* a doplňujícím komentářem rozděleném na výhody, nevýhody, jiné postřehy a shrnutí. Aby bylo skóre viditelné, musí splnit podmínku alespoň pěti hodnocení. Jelikož recenze nejsou oproti prostému hodnocení nijak váhově zvýhodněny a představují tak pouze jen jinou variantu hodnocení, je celkové skóre získáno prostým průměrem nad množinou hodnocení a recenzí.

Počet kontextů	Konstantní (4)
Stárnutí příspěvků	Nezjištěno
Uživatelská karma	Ne
Množství příspěvků	Ne
Vzorec výpočtu celkového skóre y	$y = \frac{1}{n + m} \left(\sum_{i=0}^m x_i + \sum_{j=1}^n x'_j \right)$ <p>kde m je počet hodnocení, n je počet recenzí, x_i je hodnota hodnocení i a x'_j je hodnocení recenze j.</p>
Použité reputační modely	Points, Reviews, Ratings
Vyhledávání podle kontextů	Ne
Nutnost registrace	Ano
Celkové skóre z kontextů	Ne

Tabulka 10: Souhrn vlastností reputačního serveru DealExtreme.

3.1.8 TESTFREAKS

Velkým problémem většiny serverů umožňujícím hodnotit nabízené zboží je nedostatečná odezva ze strany uživatelů. Pokud je navíc nabídka velmi široká a často se mění (sezónní zboží jako například oblečení či boty), nebude mít většina zboží hodnocení žádné a to zbývající jich bude mít tak málo, že celkové skóre nebude mít žádnou vypovídající hodnotu. Tento problém se snaží řešit švédský server *TESTFREAKS* [17], zabývající se hodnocením elektroniky. Tento web sám nic nenabízí, má pouze informativní charakter.

Server funguje tak, že sbírá data o zboží a jejich hodnocení z jiných serverů umožňujících objekty rovněž hodnotit a agreguje je do jednoho super hodnocení (tzv. *FreakScore*). Systém sbírá jak hodnocení amatérských uživatelů, tak i výsledky profesionálních recenzí psaných experty. V případě, že je výsledkem recenze hodnocení v nějaké skalární podobě, je výsledek recenze také převzat a do celkového skóre započítán. Jestliže mají dokonce nějaká přebíraná hodnocení mnoha-kontextový charakter, jsou přebírány a agregovány i tyto dílčí výsledky, nicméně na celkové skóre vliv žádný nemají.

Způsob výpočtu *FreakScore* je poměrně robustní, ale ne zcela transparentní. Na celkovém výsledku se podílí tyto faktory:

- Upřednostnění důvěryhodných zdrojů.
- Upřednostnění vyššího množství hodnocení.

- Upřednostnění expertního hodnocení před uživatelským.
- Stáří objektu.

Počet kontextů	Variabilní (2-?)
Stárnutí příspěvků	Ano (ovlivňuje hodnocení)
Uživatelská karma	Ne
Množství příspěvků	Ano
Vzorec výpočtu celkového skóre y	$y = \left(\frac{\sum_{i=0}^m x_i v_i w_i + \sum_{j=0}^n x'_j v'_j w'_j}{\sum_{i=0}^m v_i w_i + \sum_{j=0}^n v'_j w'_j} \right) * f(m + n)$ <p>kde m je počet uživatelských recenzí od určitého data, n je počet všech expertních recenzí od určitého data (omezením recenzí od určitého data zohledňuje stáří produktu), x je hodnota uživatelského hodnocení, x' je hodnota expertního hodnocení, v je váha důvěryhodnosti uživatelského zdroje, v' je váha důvěryhodnosti expertního zdroje, w je váha uživatelského hodnocení, w' je váha expertního hodnocení a funkce $f()$ ovlivňující celkové skóre na základě počtu všech hodnocení.</p>
Použité reputační modely	Reviews, Ratings
Vyhledávání podle kontextů	Ne
Nutnost registrace	Ne
Celkové skóre z kontextů	Ne

Tabulka 5 : Souhrn vlastností reputačního serveru TESTFREAKS.

3.1.9 Hotel Thailand

Server *Hotel Thailand* [22] slouží jako cestovatelův průvodce po ubytovacích zařízeních v Thajsku. Každý účastník, který si zaregistroval ubytování skrze tento systém, dostane v poslední den svého ubytování příležitost ubytování ohodnotit. Skrze odkaz v e-mailu je uživatel přesměrován na hodnotící stránku, kde může hotel hodnotit celkem z osmi aspektů: *ložnice, koupelna, čistota, potraviny a nápoje, vybavení, služby, poloha, kvalita / cena*. Jednotlivé kontexty mají stejnou váhu, takže celkové skóre hotelu je pak vypočítáno obyčejným průměrem. Jelikož množství hodnocení každého hotelu nehraje žádnou roli, slouží pro absolutní řazení nejlepších hotelů pouze jejich celkové skóre.

Počet kontextů	<i>Konstantní (8)</i>
Stárnutí příspěvků	<i>Ne</i>
Uživatelská karma	<i>Ne</i>
Množství příspěvků	<i>?</i>
Vzorec výpočtu celkového skóre y	$y = \frac{1}{m} \sum_{j=0}^m x_i$ <p><i>kde x značí hodnotu kontextu i a m je číslo kontextu.</i></p>
Použité reputační modely	<i>Reviews</i>
Vyhledávání podle kontextů	<i>Ne</i>
Nutnost registrace	<i>Ne</i>
Celkové skóre z kontextů	<i>Ano</i>

Tabulka 11: Souhrn vlastností reputačního serveru Hotel Thailand.

3.1.10 What car?

Server *What car?* [23] se zabývá hodnocením automobilů. Každý automobil v databázi má svoji vlastní kartu, kde kromě odborné recenze vytvořené profesionálem, může svoji recenzi přidat i čtenář. Na rozdíl od profesionální recenze, ve které se hodnotí auto 1-5 hvězdami celkem v devíti aspektech, může uživatel vybírat pouze ze tří vedlejších: *řízení*, *provozní náklady* a spojený kontext zabývající se *kvalitou*, *praktičností* a *řízením*, a jedním hlavním *souhrnným* kontextem. Každý kontext je pak na základě speciálního algoritmu vypočítán ze všech hodnocení zvlášť.

Počet kontextů	Konstantní (3)
Stárnutí příspěvků	Ne
Uživatelská karma	Ne
Množství příspěvků	Ne
Vzorec výpočtu celkového skóre y	Empirickým výzkumem se nepodařilo odhadnout.
Použité reputační modely	Reviews
Vyhledávání podle kontextů	Ne
Nutnost registrace	Ano
Celkové skóre z kontextů	Ne

Tabulka 12 : Souhrn vlastností reputačního serveru What car ?

3.1.11 Web Hosting Geeks

Web Hosting Geeks [24] je online reputační web zabývající se hodnocením web hostingů. K napsání recenze zde není nutnost registrace. Hodnotit můžeme až v 9 kategoriích: *technická podpora, podpora zákazníka, software, přenos dat, spolehlivost a provozuschopnost, diskový prostor, cena, kvalita a uživatelská přívětivost* jednou až pěti hvězdami. Celkové skóre je získáno jednoduchým průměrem z hodnocení recenzí, kde samotná hodnocení jsou taktéž získána jednoduchým průměrem z hodnot z výsledků jednotlivých kontextů, které jsou taktéž získány jednoduchým průměrem z dílčích recenzí.

Počet kontextů	Konstantní (9)
Stárnutí příspěvků	Ne
Uživatelská karma	Ne
Množství příspěvků	Ne
Vzorec výpočtu celkového skóre y	$y = \frac{1}{m} \sum_{j=0}^m \left(\frac{1}{n} \sum_{i=0}^{n_j} x_{ji} \right)$ <p>kde m je počet hodnocení, n je počet kontextů a x_{ij} značí hodnotu kontextu n z hodnocení m.</p>
Použité reputační modely	Reviews, This-or-Thar Voting (Byla tato recenze užitečná?), Report Abuse
Vyhledávání podle kontextů	Ne
Nutnost registrace	Ne (Nelze se zaregistrovat)
Celkové skóre z kontextů	Ne (K výpočtu celkového skóre slouží samotné kontexty)

Tabulka 13: Souhrn vlastností reputačního serveru Web Hosting Geeks.

3.1.12 BOOKING

Server BOOKING.COM [28] se zabývá on-line rezervací hotelů po celém světě. Mnoha-kontextový model umožňuje hodnotit hotely celkem v šesti aspektech - *čistota, pohodlí, poloha, služby, personál a poměr kvalita / cena*. Co však dělá tento server speciálním je, že si všímá rozdílných potřeb jednotlivých druhů návštěvníků a proto umožňuje hodnotitelům se při vytváření recenze zařadit do jedné ze šesti specifických skupin: *rodiny s malými dětmi, rodiny se staršími dětmi, skupiny přátel, starší páry, mladé páry a jednotlivci*. Při vyhledávání pak dostáváte na výběr, zda chcete vypočítat celkové skóre vytvořené všemi typy uživatelů (základní nastavení) či pouze konkrétní skupinou.

Aby mohl člověk k hotelu napsat svoji recenzi, musí se v daném hotelu nejdříve ubytovat. Na konci ubytování je pak dána člověku možnost shrnout pozitivní či negativní postřehy v recenzi doplněné o hodnocení vytvořené z výše zmíněných kontextů a zařadit se při tom do jedné z hodnotících skupin. Hodnocení recenze je vytvořeno jednoduchým průměrem z hodnot jednotlivých kontextů. U každé recenze je pak ale bohužel zobrazeno jen celkové hodnocení recenze. To znamená, že není poznat, kdo jak který kontext hodnotil. Dílčí hodnoty kontextů jsou navzájem pouze jednoduše průměrovány a zobrazeny pro všechny recenze naráz.

Počet kontextů	Konstantní (6)
Stárnutí příspěvků	Ne
Uživatelská karma	Ne
Množství příspěvků	Ne
Vzorec výpočtu celkového skóre y	$y = \frac{1}{m} \sum_{j=0}^m \left(\frac{1}{n} \sum_{i=0}^{n_j} x_{ji} \right)$ <p>kde m je počet hodnocení, n je počet kontextů a x_{ji} značí hodnotu kontextu i z hodnocení j.</p>
Použité reputační modely	Reviews, This-or-Thar Voting
Vyhledávání podle kontextů	Ne
Nutnost registrace	Ne
Celkové skóre z kontextů	Ne

Table 14: Souhrn vlastností reputačního serveru BOOKING.

3.2 Porovnání mnoha-kontextových modelů

Ze srovnání všech zkoumaných reputačních modelů uvedených v tabulkách 15 a 16 lze vyvodit několik zajímavých závěrů. Servery zabývající se širším spektrem různých druhů produktů (*Nike*, *Best Buy*) obsahují u kategorií proměnný počet kontextů, nicméně tyto kontexty u nich mají pouze informativní charakter. Při stanovování hodnocení recenze ani skóre objektu ale nehrají roli.

Z pohledu komplexnosti vychází jako vítěz reputační model serveru Fajnsmekr. Jako jediný podporuje podle řazení podle kontextů, kontexty mají při stanovení hodnocení různou váhu a při celkovém skóre objektu je brána v potaz i karma autora recenze.

Vysvětlivky k tabulkám 15 a 16:

- x - model má takovou vlastnost,
- ? - jednoznačně se nepodařilo určit, zda-li takovou vlastnost má či nemá.

Sever	Kontexty					Příspěvky			Registrace	Průměr	
	Variabilní	Konstantní	Vyhledávání	Složené skóre	Různá váha	Stárnutí	Karma	Množství	Povinnost	Jednoduchý	Vážený
Best Buy	x (2-4)							?	x	x	
Nike	x (3-5)							x	x	x	
Fajnšmekr		x(4)	x	x	x	x	x	?	x		x
Crest		x(5)		x		x			x		x
Restaurantica		x(3)				x		x	x	x	
NetTravel		x(7)		x					x	x	
Deal Extreme		x(4)				?			x	x	
Test Freaks	x(2-?)					x		x			x
Hotel Thailand		x(8)		x				?	x	x	
What car?		x(3)							x	?	?
Web Hosting Geeks		x(9)		x						x	
BOOKING		x(6)		x						x	

Tabulka 15: Přehled vlastností reálných více-kontextových reputačních systémů.

Sever	Zjištěné reputační modely					
	Reviews	Ratings	This-or-That Voting	Points	Report abuse	Vote to Promote
Best Buy	x		x			
Nike	x		x			
Fajnšmekr	x					
Crest	x					
Restaurantica	x		x		x	
NetTravel	x					
Deal Extreme	x	x		x		
Test Freaks	x	x				
Hotel Thailand	x					
What car?	x					
Web Hosting Geeks	x		x		x	
BOOKING	x	x				

Tabulka 16: Přehled vlastností reálných více-kontextových reputačních systémů.

4 Návrh reputačního modelu a aplikace

Hlavní tématem této kapitoly je na základě neformální specifikace a studia reálných modelů reputace návržení budoucí podoby vlastního mnoha-kontextového reputačního modelu a s tím spojeným návrhem funkcí pro výpočet karem uživatelů, hodnocení recenzí a celkových skóre objektů.

V menší míře je také dán prostor návrhu databázového schématu, základnímu případu užití a rozdělení uživatelských rolí. V samém závěru této kapitoly jsou stručně popsány technologie, jež bude aplikace využívat, v jaké míře jsou tyto technologie využívány a proč byly vybrány zrovna tyto technologie.

4.1 Neformální specifikace

V této kapitole si pomocí neformálního popisu nadefinujeme základní požadavky na systém pro více-kontextové hodnocení výrobků, služeb, obchodů a jiných (dohromady dále jen objekty) spojený s jejich pohodlným vyhledáváním dle preferencí uživatelů.

Systém bude umožňovat správu uživatelů a bude moci přiřazovat jednotlivým uživatelům jejich role. Základními třemi rolemi budou uživatelé *host*, *registrovaný uživatel* a *administrátor*. Tyto role se kromě *hosta* mohou ještě dále rozdělovat, například podle pravomocí *administrátorů*. Vztah s množstvím pravomocí mezi rolemi by se dal popsat jako: *host* < *registrovaný* < *administrátor*. To znamená, že registrovaný uživatel je jako host s rozšířenou pravomocí. Obdobně je definovaný vztah mezi *registrovaným uživatelem* a *administrátorem*.

Host bude mít právo v systému libovolně prohlížet veškeré informace týkající se objektů (recenze, komentáře, komentáře k *recenzím*) a bude mít i právo částečně prohlížet údaje o *zaregistrovaných uživateli* (zobrazit například seznam všech uživatelských *recenzí*). *Host* bude mít také právo hodnotit ostatní *recenze* zaregistrovaných uživatelů, kde *hodnocení recenze* bude mít podobu přidání záporného či kladného hlasu - *palce*.

Zaregistrováním se do systému a přihlášením, se stane z *hosta* zaregistrovaný uživatel. Při registraci uživatel vyplní povinně svoje přihlašovací jméno, heslo a e-mail. K hlavním pravomocím zaregistrovaných uživatelů bude patřit vytváření recenzí jednotlivých objektů a možnost přidávání komentářů k *recenzím*. Ke každému objektu bude moci uživatel napsat maximálně jednu svoji *recenzi*. Žádné komentáře a ani vlastní recenze po vložení již později nebude možno autorem měnit. Určité akce (vytváření recenzí) registrovaného se sčítají a budou vytvářet uživatelskou *participation karmu*. *Hodnocení recenzí* ostatními uživateli bude naopak vytvářet uživatelskou *quality karmu*, jež se bude počítat pro každou kategorii objektů zvlášť. Úroveň karmu jednotlivých uživatelů bude dostatečným způsobem zvýrazněna. Od určité úrovně *quality karmy* budou moci uživatelé přidávat i vlastní objekty k *recenzím*.

Mezi výhradní práva *administrátorů* bude patřit správa kategorií a podkategorií objektů, blokování uživatelů, cenzurování recenzí a komentářů. Dále budou odpovědní za vytváření hodnotících *kontextů* jednotlivých kategorií objektů, jejich váhovém podílu na *hodnocení* a budou také odpovědní za stanovování parametru ovlivňující stárnutí objektu.

Systém bude vytvářet na základě nějakého *vyhodnocovacího algoritmu* žebříčky v rámci svých kategorií objektů i napříč celým systémem. *Algoritmus* bude při vytváření *skóre* objektu uvažovat: *hodnocení* objektů z *recenzí*, dílčí *palcové hodnocení* recenzí ostatními *registrovanými* uživateli a *quality karmy* autora *recenze*. *Vyhodnocovací algoritmus* bude taktéž zohledňovat stárnutí objektů.

Systém bude umožňovat vyhledávat podle nejrozumnějších kritérií napříč danými kategoriemi a podkategoriemi s možností vyhledávat i podle kontextů dané kategorie objektů. Systém bude také umožňovat zobrazovat žebříčky týkající se uživatelů, jako třeba *uživatelé s nejvíce recenzemi* či nejvíce diskutující uživatelé.

Systém bude implementován jako webová aplikace, přístupná skrze internetový prohlížeč, která bude komunikovat s databází.

4.2 Návrh vlastního mnoha-kontextového modelu reputace

Na základě mnoha-kontextových modelů reputace použitých u sledovaných webových serverů popsaných v kapitole "Mnoha-kontextové modely v praxi" zde nadefinujeme podobu vlastního mnoha-kontextového modelu. Jelikož se náš server nebude zabývat hodnocením konkrétní množiny objektů, ale bude umožňovat hodnotit prakticky cokoliv, nepřípadá statická množina kontextů společná všem oblastem v úvahu. V případě proměnného počtu kontextů se tedy budeme spíše inspirovat modelem obchodního serveru *BestBuy* [12].

Mezi další vlastnosti, které by náš model rozhodně mít měl, navzdory tomu, že jsou využívány minimálně, je vyhledávání podle kontextů a jejich různá váha. Když už náš model bude podporovat více-kontextové hodnocení, byla by chyba neposkytnout i možnost podle těchto kontextů vyhledávat či řadit. Bez této možnosti by byli uživatelé odkázáni pouze na řazení podle celkového skóre objektu. S uvažováním váhového rozdílu kontextů se nám pak otvírá možnost definování hlavních a vedlejších kontextů, které by si jinak byly rovny.

V otázce stanovování hodnocení recenze je situace mezi sledovanými modely vyrovnaná. Přibližně půlka modelů získá celkové skóre recenze výpočtem z hodnocení jednotlivých kontextů. Náš model bude postupovat stejně. Tento výpočet je daleko více transparentní, navíc se tím pádem předchází řadě nelogických hodnocení a chyb plynoucích z dvojího navzájem nezávislého hodnocení vedlejších kontextů objektu a hlavního hodnocení. V takovém případě klidně může dojít k situaci, kdy

všechny kontexty mohou mít minimální hodnotu, ale hlavní hodnocení, a tím pádem i hodnocení recenze, bude maximální.

Přestože většina sledovaných modelů karmu uživatelů nějakým způsobem zaznamenávala, při hodnocení objektů ani u řazení ale nebyla uvažována. Náš model bude v tomto směru značně pokrokový a bude uvažovat jak *quality karmu* recenze, tak *quality karmu* uživatele. Jelikož i počet palců u každé recenze je důležitým ukazatelem, bude uvažována i *p-karma recenze*. Při výpočtu celkového skóre naopak nebude uvažována zbývající *p-karma* uživatele. Ta bude mít v modelu pouze informativní charakter.

Aby náš model správně reagoval na stárnutí objektů a jejich recenzí, bude počítat s jejich postupnou degradací. Pro potřeby našeho modelu bude dostatečné uvažování recenzí v rámci časového intervalu, jež ono pozvolné upadání v průběhu času zajistí.

Jelikož získání celkového skóre objektu bude v našem modelu klíčové, věnuje se mu podrobně celá následující kapitola.

4.3 Návrh výpočtu celkového skóre objektu

Jelikož celkové skóre objektu bude získáno z hodnocení dílčích recenzí a jejich počtu v platném časovém pásmu, bude nutné nejprve spočítat hodnocení samotných recenzí. Výpočet každé dílčí recenze bude ale poměrně komplexní operací, protože vliv na hodnocení budou mít celkem 4 faktory - *q-karma* hodnotícího uživatele (autora recenze), *q-karma* recenze, *p-karma* recenze a výsledné hodnocení recenze samotné vytvořené autorem recenze. Před návrhem funkce pro výpočet celkového skóre objektu tak musí být navrženy všechny pomocné funkce pro získání potřebných faktorů.

Pro potřebu snadné srozumitelnosti, usnadnění dalšího zpracování (zejména ve funkcích shrnující výsledky) a jednoduchého zpětného převodu na libovolnou podobu budou všechny hodnoty *q-karmem* normalizovány na hodnotu z intervalu $< 0, 1 >$. Stejná podmínka bude platit i pro návratové hodnoty všech dále navržených pomocných funkcí.

4.3.1 Stanovení střední hodnoty *q-karmy*

Před definicí dílčích metod výpočtu bude nejprve nutné v rámci intervalu $< 0, 1 >$ stanovit jakousi střední neutrální hodnotu *q-karmy*. Podmínkou pro tuto hodnotu je dobrá reprezentace dostatečného rozdílu oproti maximální i minimální hodnotě *q-karmy*. Jako první vhodná hodnota se jeví číslo 0.5. Tato hodnota však dost dobře neumožňuje růst v rámci daného intervalu, neboť je jen 2x menší než maximální hodnota. Z důvodů stanovení desetinásobného rozdílu neutrální a maximální *q-karmy* bude jako střední hodnota určeno číslo 0.1. Nejnižší možnou hodnotou, jakou může *q-karma* nabývat, bude

desetina neutrální hodnoty q-karmy a to hodnota 0.01. Interval $< 0, 0.01 >$ nebude uvažován a v případě jakékoliv hodnoty z tohoto intervalu bude brána jeho maximální hodnota.

Hlas uživatele s maximální pozitivní q-karmou 1 tak bude mít stejnou váhu jako 10 uživatelů s q-karmou základní a váhu jako 100 uživatelů s q-karmou maximální negativní. Hodnota střední q-karmy bude přidělena novým uživatelům při registraci.

4.3.2 Q-karma uživatele

Quality karma uživatele bude sloužit jako první ze tří vah nutná pro výpočet hodnoty recenze. Pro výpočet q-karmy uživatele bude sloužit hodnocení jeho akcí (psaní recenzí) od ostatních uživatelů. Váhy palců jednotlivých uživatelů si budou při stanovování úrovně q-karmy uživatele rovny. To znamená, že v tomto případě bude váha hlasu anonymního uživatele rovna váze hlasu dlouholetého člena s vysokou reputací. Úvaha standardním způsobem váhově zvýhodňovat hlasy hodnotících uživatelů není možná, neboť zde významně hrozí zneužití vysoké váhy hlasu mezi "kamarády".

Pro potřeby vnitřního výpočtu bude nutné nalézt 2 křivky a k nim odpovídající 2 rovnice. První křivka bude reprezentovat pozitivní průběh q-karmy definovaný na intervalu $(0.1, 1 >)$. Od křivky lineární na stejném intervalu se stejnými hraničními body se musí lišit zprvu rychlejším růstem, aby zpočátku motivovala nové uživatele a na konci naopak zpomalením, aby nebylo možné dosáhnout maxima tak snadno.

Druhá křivka bude naopak na intervalu $< 0.01, 0.1 >$ reprezentovat průběh negativní q-karmy. Na rozdíl od křivky lineární musí mít zpočátku pomalejší sestupnou tendenci, aby zprvu nové uživatele příliš nepenalizovala, a musí zasáhnout až v případě značného množství záporných hlasů.

Empiricky tak byly stanoveny pro výpočet q-karmy v rozmezí 1 až 100 palců 2 rovnice:

$$qK^{+u} = \left(\frac{x}{3x + 20} \right) \left(\frac{\ln(\sqrt{x})}{0.794} \right) + 0.1 \quad (11)$$

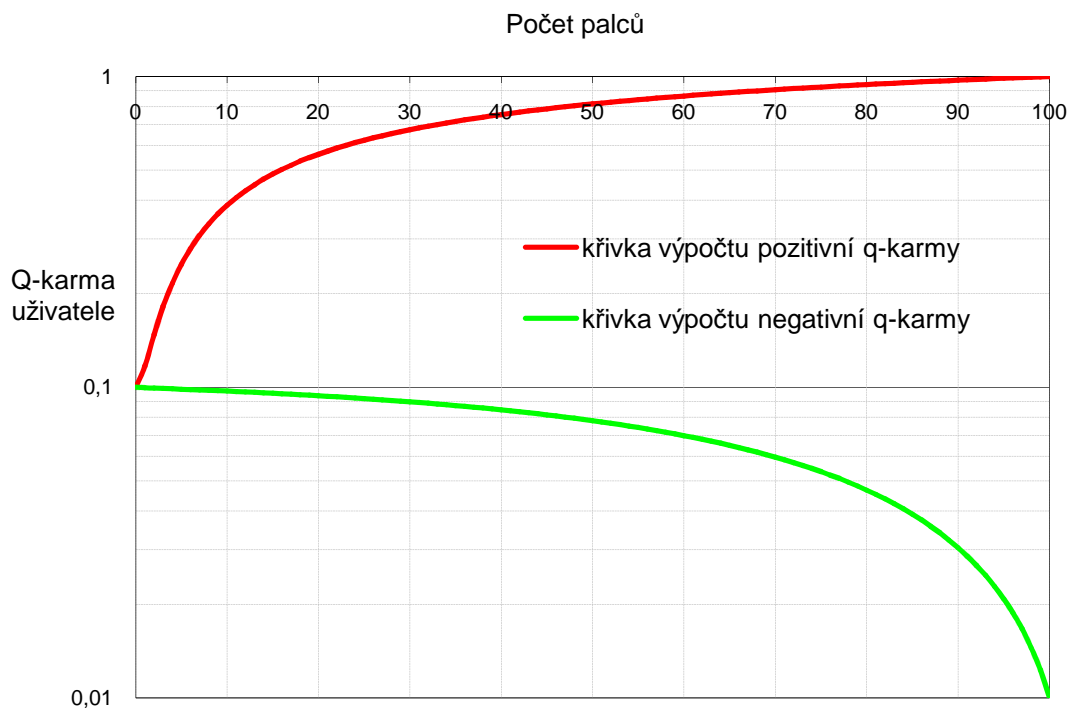
kde:

- qK^{+u} je výsledná kladná hodnota q-karmy,
- x je počet kladných palců a

$$qK^{-u} = \frac{e^{\frac{x}{43.9}}}{100} + 0.11 \quad (12)$$

kde:

- qK^{-u} je výsledná záporná hodnota q-karmy,
- x je počet záporných palců.



Obrázek 3: Průběh funkce výpočtu uživatelské q-karmy.

4.3.3 Q-karma recenze

Druhým faktorem ovlivňujícím celkové skóre objektu je kvalitativní úroveň recenze samotné. Jelikož v případě stanovení q-karmy recenze budeme uvažovat i váhu jednotlivých palců vztažené od q-karmy hodnotících uživatelů, nebude v tomto případě možné stanovit q-karmu pouze z počtu palců. Pro každou recenzi bude nutné nejdříve seskupit množinu jejich palců podle kladné a záporné hodnoty palce a takto sečtené q-karmy uživatelů odečíst a výsledek podělit hodnotou neutrální q-karmy:

$$p = \frac{\sum_{i=0}^n w_i - \sum_{j=0}^m v_j}{0.1} \quad (13)$$

kde:

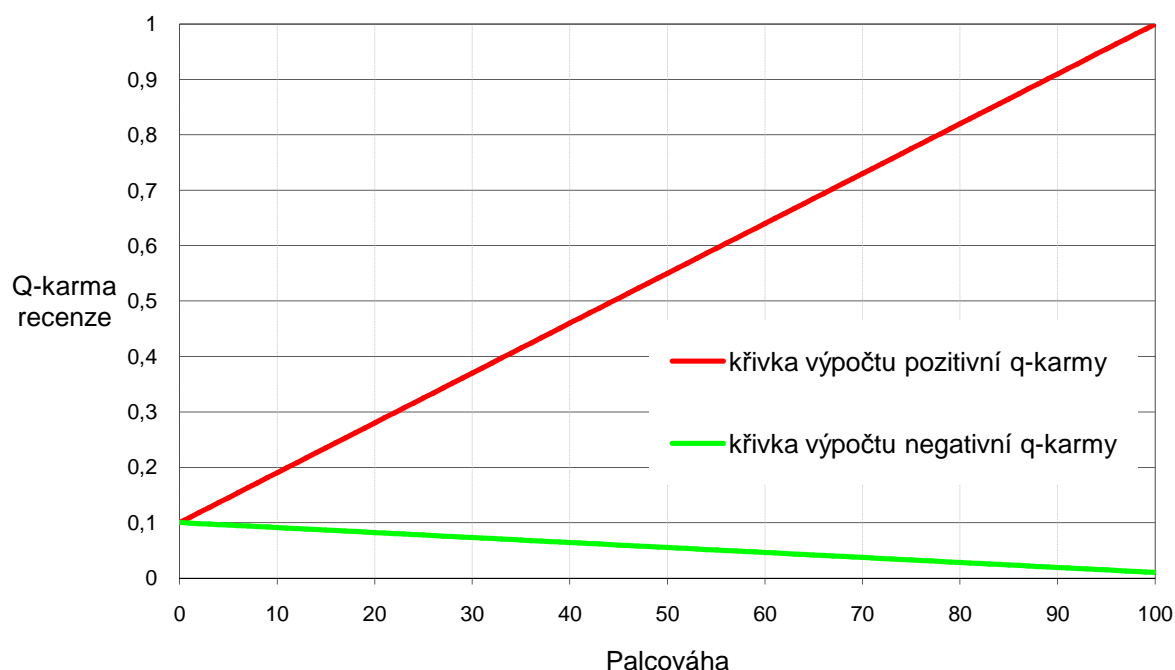
- n je počet pozitivních palců,
- m je počet negativních,
- w_i je q-karma uživatele i dávajícího kladné hodnocení,
- v_j je q-karma uživatele j dávajícího záporné hodnocení,
- p bude výsledná hodnota tzv. *palcováhy* (jelikož v tomto případě bude nabývat hodnota počtu palců hodnot z oboru reálných čísel, nedá se zde již mluvit o počtu palců).

Pro výpočet kladné a záporné q-karmy recenze z palcováhy zde bude opět nutné představit 2 rovnice. Počáteční a koncové body budou stejné jako v případě výpočtu q-karmy recenze avšak v tomto případě nám bude stačit k výpočtu funkce lineární. Na základě empirického výzkumu tak bude představena následující funkce:

$$qK^r = \begin{cases} -0.0009|p| + 0.1, & p < 0 \\ 0.009|p| + 0.1, & p > 0 \\ 0.1, & p = 0 \end{cases} \quad (14)$$

kde:

- qK^r je výsledná hodnota q-karmy recenze,
- x je hodnota palcováhy na intervalu $< 0, 100 >$.



Obrázek 4: Lineární funkce pro výpočet q-karmy recenze.

4.3.4 P-karma recenze

Posledním faktorem ovlivňujícím váhu recenze, který bude taktéž potřeba při výpočtu celkového hodnocení recenze vzít v úvahu, je množství palců každé recenze.

Uvažujme 2 recenze, kde první má vysoké uživatelské hodnocení, ale pouze 3 palce libovolné hodnoty, zatímco druhá má hodnocení nízké, ale palců má 300. Z minimálního počtu hodnocení první recenze a velkého rozdílu oproti druhé vyplývá, že druhá recenze má více informací k výpočtu, daleko lépe odráží názor uživatelů a zároveň minimalizuje riziko náhody a měla by proto mít větší váhu.

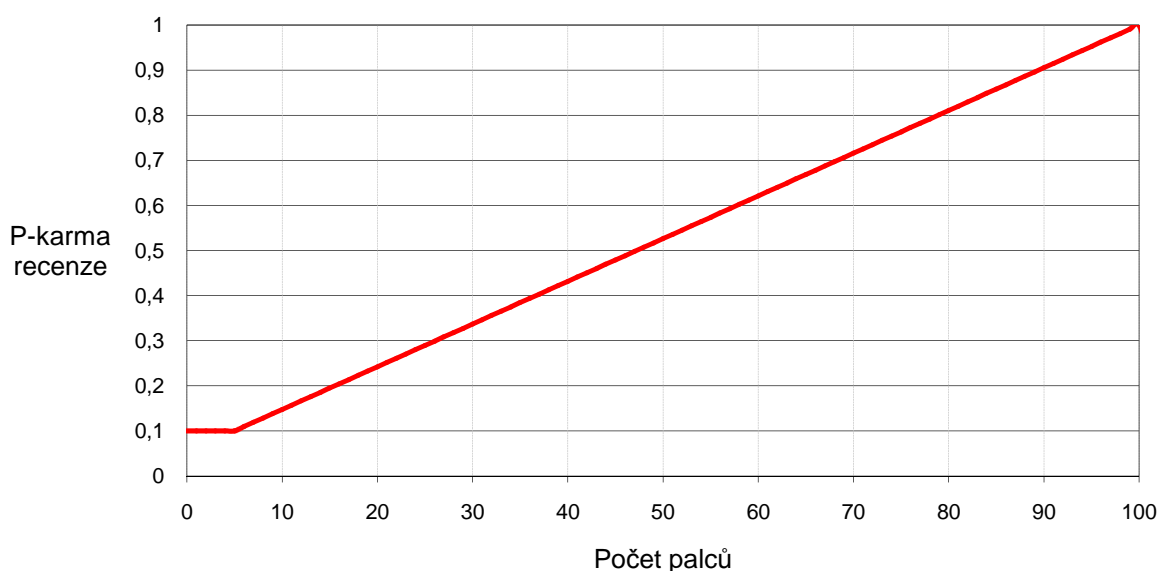
U první recenze si naopak kvůli nedostatečnému množství palců ostatních uživatelů nemůžeme být získaným výsledkem jisti a proto bychom (do překročení jisté hranice množství palců) měli takovéto recenzi přiřadit minimální váhu nebo ji neuvažovat vůbec. V extrémním případě by totiž nová recenze s nejvyšším hodnocením a jedním kladným palcem přeskočila všechny ostatní a byla nejvýznamnějším prvkem výpočtu. Takovéto chování je však nežádoucí.

Jako základní počet palců, od kterého se bude váha počtu palců započítávat do celkového hodnocení recenze, bude stanoveno na 5 hodnocení. Tato fixní hodnota bude všem kategoriím společná. Aby měli všechny váhy hrající roli v celkovém hodnocení recenze stejnou neutrální a maximální váhu, bude i křivka reprezentující výpočet p-karmy recenze vycházet z neutrální hodnoty 0,1 a končit maximem v 1. Jelikož na křivku nejsou kladeny žádné další speciální požadavky, byla empirickým výzkumem odvozena výsledná podoba v následujícím tvaru:

$$pK^r = \begin{cases} 0,1, & 0 \leq x \leq 5 \\ \frac{0,9(x-5)}{95} + 0,1, & 5 < x \leq 100 \\ 1, & x < 100 \end{cases} \quad (15)$$

kde:

- pK^r je výsledná hodnota p-karmy recenze,
- x je počet palců.



Obrázek 5: Průběh výpočtu p-karmy recenze.

4.3.5 Hodnocení recenze

Aby mohl být nadefinován výpočet celkového skóre objektu, zbývá vypočítat nejdůležitější věc, a to hodnocení recenze. Doposud uváděné funkce tuto hodnotu pouze váhově zvýhodňovaly (srážely). Hodnocení recenze se získá při vytváření recenze autorem při stanovování hodnot kontextů. Způsob výběru hodnot kontextů bude vizualizován pomocí select boxu a výsledek se bude průběžně aktualizovat. Uživatel dostane na výběr z hodnot 0 až 5 , kde jedna znamená nejhorší a pětka nejlepší známku. Pro použití ve výpočtu bude nutné dílčí hodnoty opět normalizovat. Hodnocení recenze r se vypočítá jako vážený průměr hodnot kontextů:

$$r = \frac{\sum_{i=0}^m (h_i v_{h_i})}{\sum_{i=0}^m (v_{h_i})} \quad (16)$$

kde:

- m je počet kontextů, pomocí kterých lze objekt hodnotit,
- h_i je hodnota kontextu i ,
- v_{h_i} je váha daného kontextu h_i přednastavená pro danou kategorii.

4.3.6 Celkové skóre objektu

Nyní, když jsme si v předchozích kapitolách nachystali pomocné funkce, můžeme konečně definovat výslednou podobu funkce pro výpočet celkového skóre objektu. Hodnota celkového skóre objektu R nabývají hodnoty z intervalu $< 0, 1 >$ se bude počítat v rámci své kategorie jako:

$$R = \frac{\sum_{i=0}^n [(r_i) * (qK_i^u * pK_i^r * qK_i^r)]}{\sum_{i=0}^n (qK_i^u * pK_i^r * qK_i^r)}, \quad (17)$$

kde:

- n je množství recenzí započítaných do celkového hodnocení (započítány jsou pouze ty recenze, jež se nachází v platném časovém intervalu),
- r_i je hodnocení recenze i ,
- qK_i^u je hodnota q-karmy uživatele recenze i v době vytvoření recenze,
- pK_i^r je p-karma recenze i - funkce zohledňující množství palců (kladných i záporných),
- qK_i^r je q-karma recenze i .

4.4 Návrh aplikace

V podkapitole návrh aplikace budou uvedeny některé základní diagramy používané v počáteční fázi návrhu projektu.

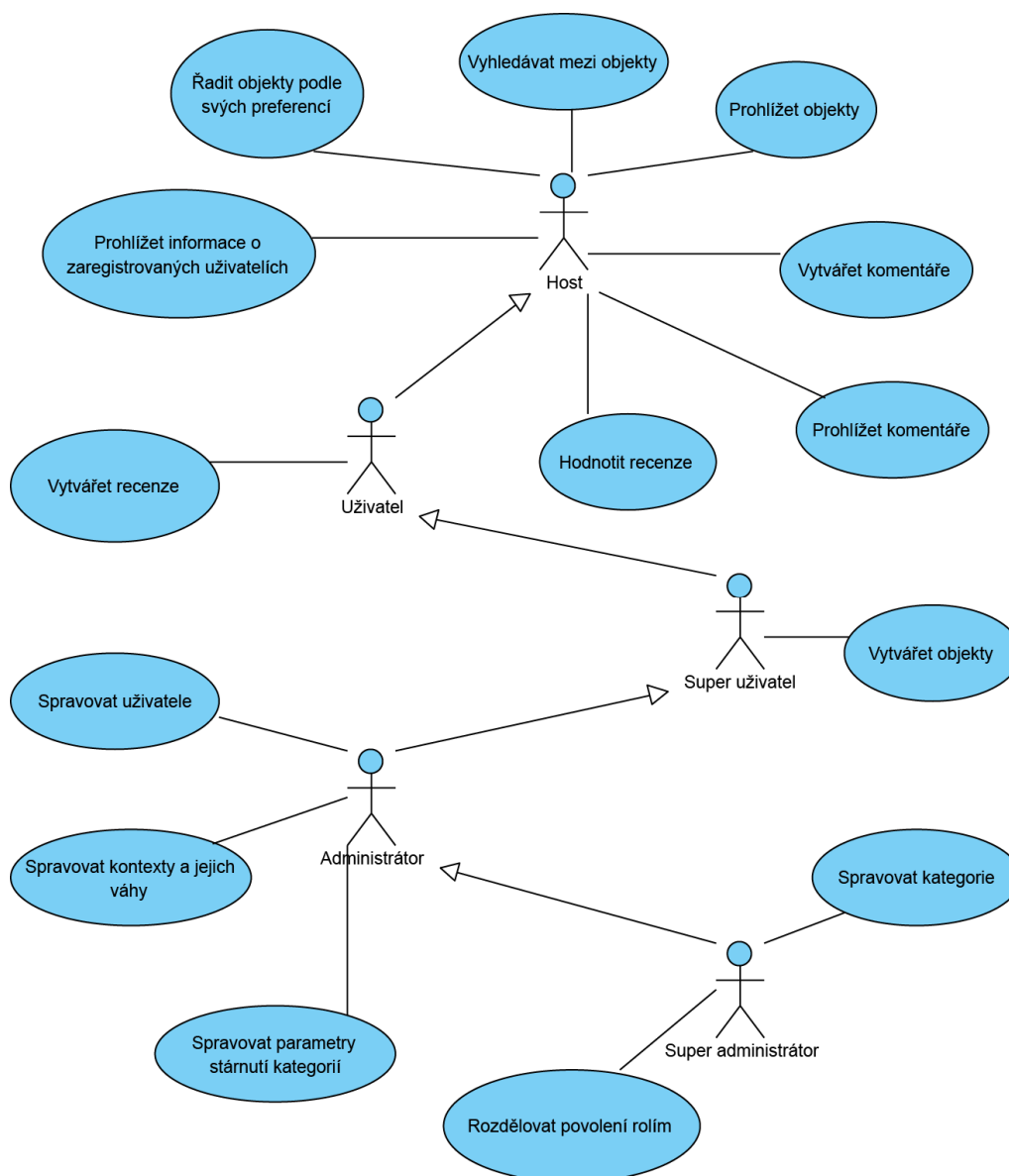
4.4.1 Diagram případu použití

Z neformální specifikace plynou následující požadavky na aktéry v systému:

- Host - reprezentuje nejobecnějšího aktéra. Komunikuje s dostupnými případy užití a to konkrétně s nemodifikujícími akcemi. Výjimku tvoří vytváření komentářů a *palcování* recenzí.
- Registrovaný uživatel - reprezentuje hlavního aktéra podílejícího se na vytváření obsahu serveru v podobě vytváření objektů a recenzí.
- Administrátor - je aktérem s nejvyššími právy v systému. Může zobrazovat a modifikovat libovolná data v systému (komunikuje se všemi případy užití).

Jelikož je množina užití administrátorů značně rozsáhlá, případ užití *vytváření objektů* je v systému klíčový a neformální specifikace je v ohledu přesného rozdělení aktérů v systému poměrně benevolentní, budeme už při návrhu počítat se dvěma dodatečnými aktéry rozdělujícími si případy užití *registrovaných uživatelů* a *administrátorů*. Registrovaný uživatel tak bude rozdělen na aktéry *uživatel* a *super uživatel* a k aktéru *administrátor* přibude navíc ještě jeho specializovanější verze v podobě aktéra *super administrátora*.

Výsledný diagram případu použití na obrázku 6 zobrazuje případy užití asociované se všemi výše uvedenými aktéry.



Obrázek 6: Diagram případu použití budoucí aplikace.

4.4.2 Návrh databázového schématu

Všichni uživatelé systému budou mít společnou tabulku DI_USER³. Jelikož každý uživatel bude moci mít vícero rolí v systému a každá role nebude mít zřejmě jen jednoho zástupce, je nutností mezi uživateli a DI_ROLE reprezentující role použití pomocné vazební tabulky. Ke každé roli pak bude příslušet množina pravomocí.

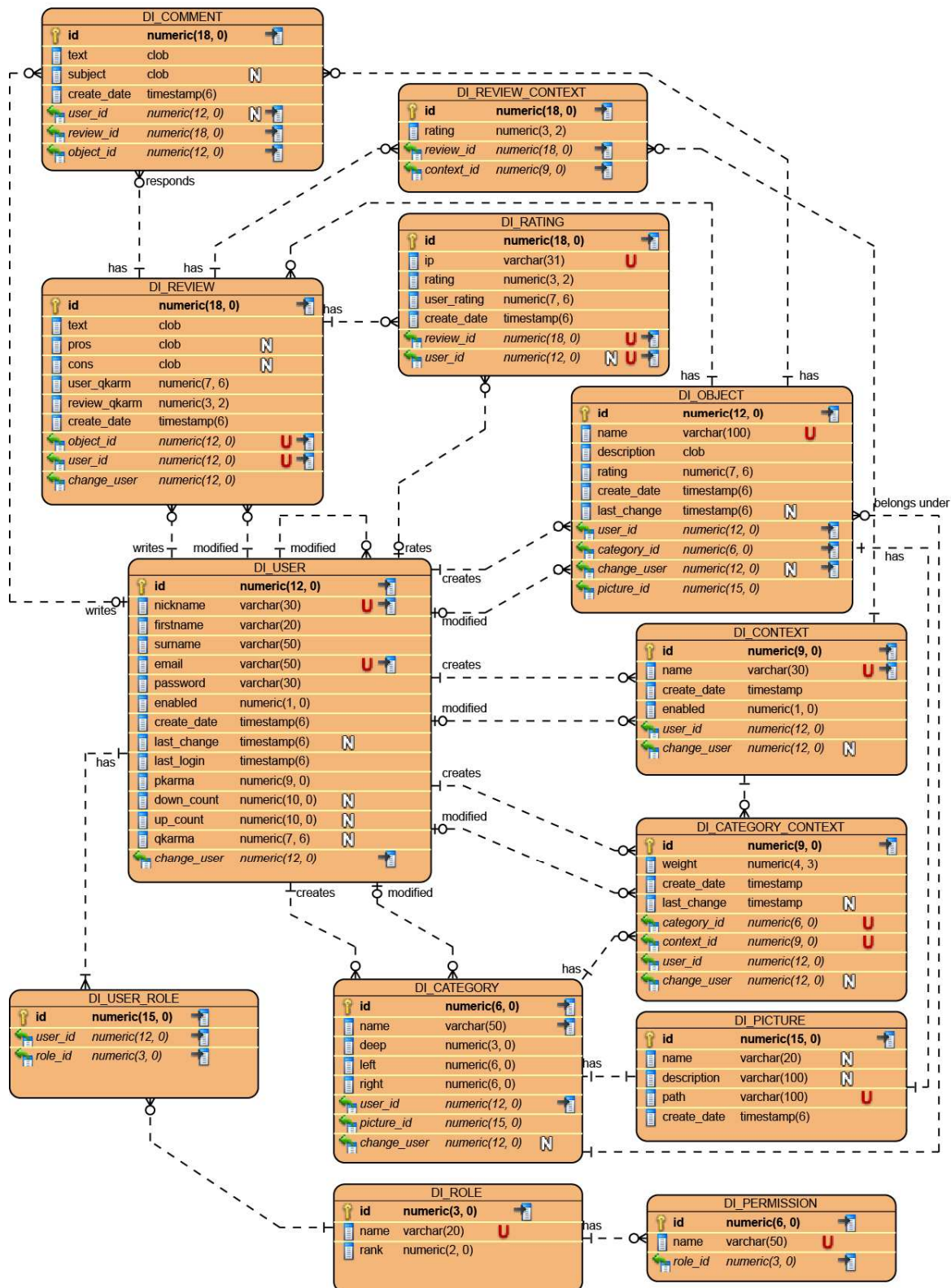
³ Prefix "DI_" u všech tabulek v databázovém schématu má nejen význam odlišení od společného jmenného prostoru se systémovými tabulkami ale hlavně také usnadňuje psaní databázových dotazů pomocí editorů s nápovědou.

Protože každá kategorie může mít více kontextů a každý kontext může být použit ve více kategoriích, bude mezi tabulkami s kategoriemi (DI_CATEGORY) a kontexty (DI_CONTEXT) opět pomocná vazební tabulka. Jelikož kontext může být použit u více kategorií, kde v každé kategorii může mít různou váhu, není možné tento údaj ukládat společně s kontextem, ale bude muset být součástí vazební tabulky DI_CATEGORY_CONTEXT. Ze stejného důvodu nebude možné u kategorií se stejnou rodičovskou kategorií použít dědičnosti a kontexty zdědit, ale budou muset být nastaveny pro každou kategorii zvlášť. Tabulka s kategoriemi bude obsahovat navíc ještě pomocné sloupce pro potřebu traverzování⁴.

Pro potřebu ukládání recenzí bude sloužit tabulka DI_REVIEW. Složený unikátní klíč objekt - uživatel značí, že každý uživatel bude moci napsat maximálně jednu svoji recenzi. Ke každé recenzi bude patřit sada hodnocení kontextů, jež se budou ukládat do tabulky DI_REVIEW_CONTEXT. K hodnocením recenzí samotných bude sloužit tabulka DI_RATING. Dvojice složených unikátních klíčů ip - review_id a user_id - review_id bude zajišťovat, že z dané IP adresy půjde ohodnotit recenzi pouze jedenkrát. Stejně omezení pak bude platit i pro přihlášeného uživatele.

Význam a funkce zbývajících tabulek je zřejmá a proto již jen ve zkratce. Do tabulky DI_OBJECT se budou ukládat veškeré objekty (produkty), tabulka DI_COMMENT bude k objektům a recenzím uchovávat komentáře a DI_PICTURE bude sloužit k ukládání fotek. Obrázek 7 znázorňuje výslednou podobu navrženého datového schématu.

⁴ Traverzování kolem stromu je technika pro efektivní práci se stromovými datovými strukturami.



Obrázek 7: Navržené databázové schéma aplikace.

4.5 Použité technologie

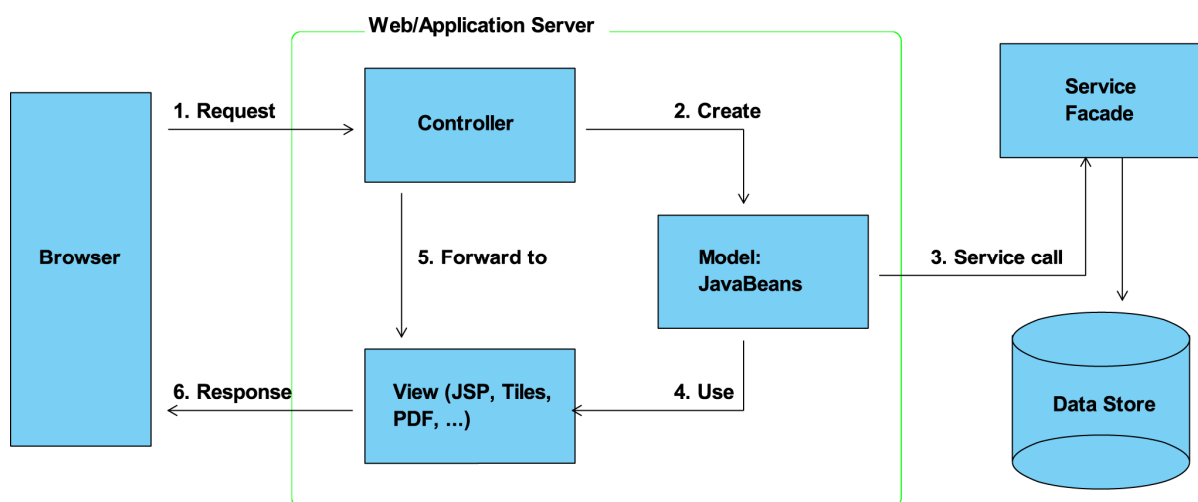
V této podkapitole je stručně popsán výčet použitých technologií a klíčových knihoven, způsob jakým jsou v projektu využívány a důvod, proč byly vybrány.

4.5.1 Spring

Spring [26] je modulární aplikační kontejner (framework) pro vývoj J2EE aplikací. Modularitou je zde myšleno využívání jen těch částí frameworku, které jsou potřeba. Jádru springu je postaveno nad využitím návrhového vzoru Inversion of Control, jež funguje na principu přesunutí zodpovědnosti za vytvoření a provázání objektů z aplikace na framework.

Nad springem je implementováno jádro celé aplikace. Aplikace využívá modulů - Core (Beans, Core), Web (Web, Servlet) a Data Access/Integration (ORM). Přestože je spring navržen tak, aby podporoval téměř všechny významné implementace modelu MVC (JSF, Struts, Tapestry), rozhodl jsem se pro výběr MVC modelu poskytovaného samotným Springem. Důvodem pro toto rozhodnutí bylo zejména rozsáhlá komunita, kvalitní dokumentace a v neposlední řadě i vzrůstající obliba nejen v České republice.

Jelikož model MVC Springu [5] hraje v aplikaci klíčovou roli, bude popsán trochu detailněji. Klíčovou třídou v tomto modelu je třída *DispatcherServlet*. Tento servlet je odpovědný za zachytávání žádostí a rozhodnutí, který *Controller* danou žádost zpracuje. Každý *Controller* má pak jako návratovou hodnotu objekt *ModelAndView*, kde *Model* není v podstatě nic jiného než jednoduchá mapa držící data v podobě JavaBeans a objekt *View*, jež v sobě nese referenci se jménem. Třída *ViewResolver* se pak postará o rozhodnutí, které *View* návratový model zpracuje a zobrazí.



Obrázek 8: Architektura modelu MVC Springu (z důvodu jednoznačnosti byly ponechány originální názvy).

4.5.2 Spring Security

Spring Security (do roku 2008 pod názvem Acegi Security) je J2EE framework poskytující pokročilou autorizaci, autentizaci a další bezpečnostní mechanismy.

Aplikace využívá tento framework pro autentizaci (přihlášení uživatelů do systému) a autorizaci (definici pravomocí uživatele).

4.5.3 Hibernate

Hibernate [27] je framework mapující pomocí anotací či konfiguračních souborů databázové relace na javové entity (ORM). Pro komunikaci s databází pak slouží objektový jazyk Hibernate Query Language (HQL).

V aplikaci slouží jako pojící vrstva mezi databází a aplikací samotnou. Pro zpřehlednění dotazů, snadnou dynamickou modifikaci a zjednodušení jejich zápisu aplikace ve velké míře využívá vyhledávacích kritérií poskytovaných aplikačním rozhraním Hibernate Criteria Queries.

4.5.4 Oracle

Oracle je systém pro řízení báze dat s pokročilými možnostmi zpracování dat, vysokým výkonem a snadnou škálovatelností.

V projektu plní roli úložiště dat. Kvůli podpoře české jazykové sady byla vybrána XE verze ukládající znaky na více bajtů.

4.5.5 Tiles

Tiles je šablonovací framework usnadňující vývoj webových uživatelských rozhraní. Umožňuje rozložit definici stránky na fragmenty a pomocí těchto fragmentů pak za běhu výslednou podobu stránky z těchto fragmentů sestavit.

Pomocí Tiles je v aplikaci přednastavena sada rozložení uživatelského rozhraní. Slouží také jako jednotné místo definice názvu pohledů pro směřování v MVC Springu.

4.5.6 JSP + JSTL

Technologie JSP společně se standardní knihovnou značek JSTL umožňuje jednoduše vytvářet dynamický obsah webu.

Sada značek standardní knihovny je využívána pro formátování a zobrazení objektů vrácených jednotlivými kontrolery springu.

4.5.7 JQuery

JQuery je JavaScriptová knihovna poskytující rozhraní pro obohacování aplikace o pokročilé efekty, animace a prvky ovládacího rozhraní umožňující vytvářet vysoce interaktivní aplikace.

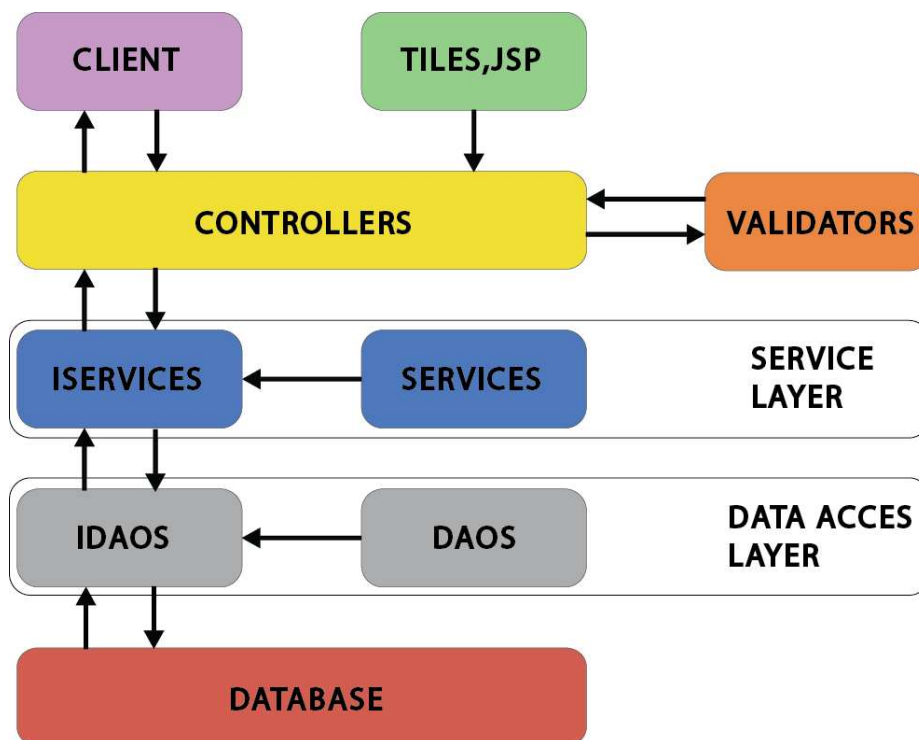
V aplikaci je tento framework využíván pro vytváření pokročilých ovládacích prvků uživatelského rozhraní a pro asynchronní komunikaci se serverem (AJAX).

5 Implementace aplikace

Smyslem kapitoly implementace není detailní popis všech tříd a jejich funkcí, ale všímá si spíše jen těch zajímavějších implementovaných konceptů a ne zcela běžných řešení. V první podkapitole je tak zmíněna architektura aplikace, následovaná podkapitolou týkající se zabezpečení. Dalším popsáním tématem je princip alternativního způsobu řazení objektů a kapitola je zakončena zmínkou o možnosti pohodlného a přehledného vytváření databázových dotazů pomocí kritérií.

5.1 Architektura aplikace

Implementace architektury staví na základním konceptu udržitelné aplikace - rozdělení do logických vrstev. Databáze je v aplikaci přístupná skrze vrstvu přístupu k datům (Data Acces Layer - DAO). Tato vrstva využívá pro komunikaci aplikačního rozhraní Hibernate. Metody DAO vrstvy zapouzdřuje Servisní vrstva. V této vrstvě je koncentrována bussines logika celé aplikace. Metody DAO a Servisní vrstvy jsou přístupné skrze rozhraní poskytované jednotlivými třídami. Metody servisních bean využívají ostatních servisních bean, ale hlavně také kontrolery, jejichž hlavní funkcí je zachycování požadavků od klienta. Pokud klient posílá nějaká data pocházející z formulářů, jsou tato data nejprve validována pomocnými validovacími beanami - validátory.



Obrázek 9: Architektura aplikace.

5.2 Autorizace a autentizace

K autentizaci a autorizaci využívá aplikace zabezpečovacího frameworku Spring security. Po úspěšném přihlášení skrze přihlašovací stránku je vytvořen nový objekt držící kromě uživatelského jména, hesla a dalších vlastností také kolekci povolení a uživatelských rolí odpovídajících danému účtu. Tato kolekce má podobu řetězcových konstant, oproti kterým jsou porovnávány zabezpečené funkce, objekty či úseky kódu v .jsp stránkách. Na implementační úrovni ale není rozdíl mezi povolením a uživatelskou rolí, neboť obě jsou tvořeny pouze řetězcovou konstantou.

Samotné zabezpečení aplikace je provedeno na 3 úrovních. První autentizační ochranou je ochrana na úrovni samotných java server pages stránek. Jednotlivé chráněné bloky kódu jsou ohraničeny speciálním bezpečnostním tagem `<sec:security />`. Ve vlastnosti `access` tohoto tagu je uloženo právo či povolení, jež musí být přítomno v kolekci povolení či rolí objektu držícího přihlášeného uživatele, aby byl vnitřní obsah chráněné oblasti zpřístupněn. V aplikaci je takového zabezpečení například využíváno při zobrazování tlačítek pro přidávání objektů, recenzí či odkazů na konfigurační obrazovky skrze uživatelské menu. Takovéto zabezpečení je však velmi nedostačující. To, že případný útočník nevidí odkaz na zabezpečenou stránku, neznamená, že na ni nemůže přistoupit přímo.

Zde tedy nastupuje druhá vlna autorizace a to zabezpečení funkcí kontrolerů či kontrolerů samotných odchytávajících jednotlivé požadavky (adresy). Chráněné funkce či třídy jsou tentokrát obalené zabezpečující anotací s parametrem obdobným jako měla vlastnost `access` zabezpečujícího security tagu. V případě, že se uživatel pokusil přistoupit přímo a nemá požadované oprávnění, je přesměrován na chybovou stránku informující uživatele o nedostatečných právech.

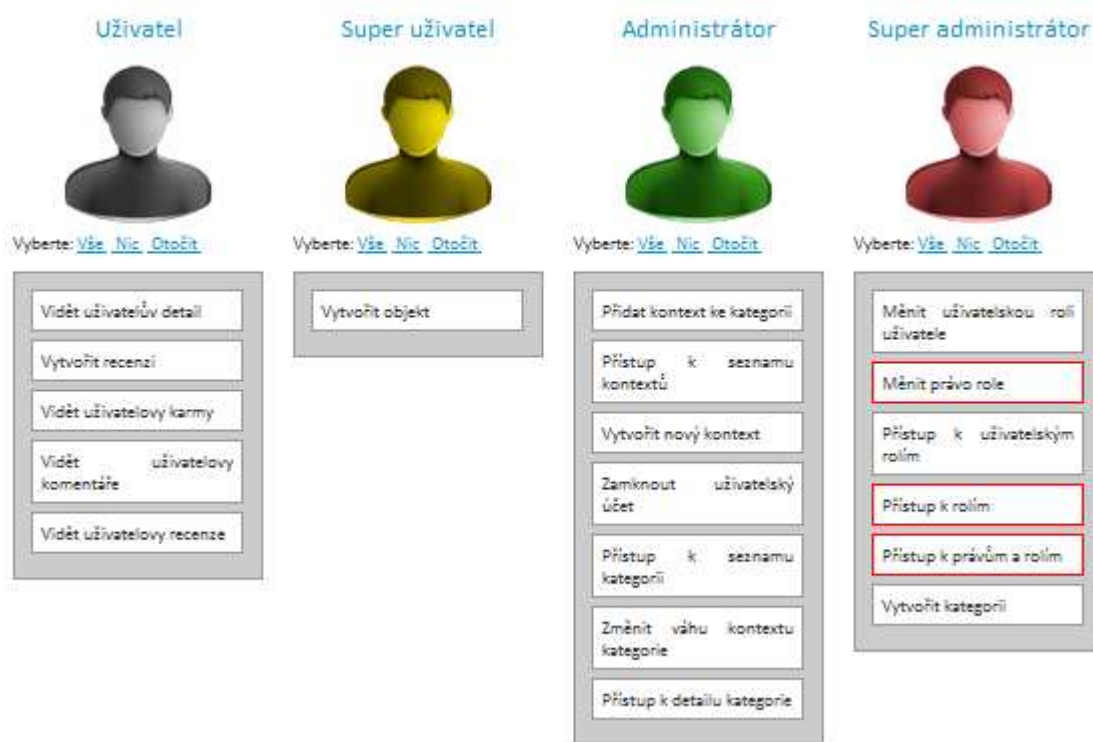
Třetí, spíše už jen doplňkovou a mající význam spíše v budoucnu, je zabezpečení servisní vrstvy. Uživatel se nyní nemůže dostat přímo k servisním funkcím, neboť tyto funkce jsou volány z kontrolerů, a tyto funkce již zabezpečené jsou. V budoucnu nicméně nemusí být vyloučena další implementace aplikace, nahrazující model MVC například architekturou klient-server pro mobilní zařízení.

Implementace aplikace počítá s tím, že s aplikací bude komunikovat celkem 5 rolí (typů) uživatelů. První rozdělení uživatelů je podle toho, zda-li je uživatel přihlášen či nikoliv. Nepřihlášený uživatel je takzvaný anonymní uživatel. Takovému to uživateli je zabezpečovacím frameworkem automaticky přidělena role `ANONYMOUS`. Po autentizaci se stává z anonymního uživatele uživatel s rolí `USER` či `ADMIN`. Každá z těchto rolí se dále dělí na 2 podrole.

Pro možnosti nastavení pravomocí jednotlivých rolí byla vyvinuta komponenta umožňující pomocí *drag & drop* pohodlně přesouvat jednotlivé pravomoci či celé skupiny pravomocí mezi uživatelskými rolemi. Z důvodu zachování přístupu k jistým nastavením je pravomoc měnit rozložení pravomocí samotných fixní a není možné ji měnit.

Při návrhu a implementaci byl kladen důraz na platnost stereotypu dědičnosti, to znamená, aby role nadřazená měla všechny pravomoci rolí podřazených. Při implementaci se vycházelo z předpokladu, že administrátor je také uživatel. Výhodou takového přístupu je, že v aplikaci tak nemůže dojít k situaci, že by obyčejný uživatel mohl vykonat něco, co nemůže samotný administrátor. Druhou výhodou je odstranění duplicitních práv v tabulce DI_PERMISSION. Nevýhodou je naopak nárůst záznamů v tabulce DI_USER_ROLE, kdy například pro každého super administrátora bude tabulka obsahovat 4 záznamy všech typů rolí pro daného uživatele. Očekává se však, že obou typů administrátorských účtů dohromady bude v řádu desítek a účtů super uživatelů bude do deseti procent počtu všech uživatelských účtů 1.

Změny v nastavení práv rolí se neprojeví na uživateli hned, ale až při jejich dalším přihlášení. Neočekává se však, že by zde docházelo často k velkým změnám.



Obrázek 10: Komponenta pro rozdělování práv rolím.

Při návrhu bylo taktéž počítáno s možností přidat další uživatelskou roli. Z důvodu lokalizace názvu role a tím pádem přímou změnou konfiguračního souboru s lokalizacemi, bylo nakonec od takového dynamického vytváření typů rolí upuštěno.

5.3 Více-kontextové vyhledávání a řazení

Pro možnost usnadnění nastavování vah jednotlivých kontextů kategorie pro potřebu řazení a vyhledávání byla vytvořena speciální komponenta vycházející z ovládacího prvku posuvníku. Tato komponenta má podobu sady posuvníků, které je možné v definici kategorie dynamicky přidávat. Hodnota každého posuvníku je ohraničena maximální a minimální hodnotou, kde suma hodnot všech těchto posuvníků je rovna konstantní hodnotě⁵. V našem případě je touto konstantní hodnotou 100%.

Posun libovolného posuvníku směrem doleva má za následek rovnoměrnou distribuci úbytku hodnoty posuvníku na ostatní posuvníky. Analogicky posun doprava způsobí rovnoměrné odčerpávání z hodnot ostatních posuvníků.

The interface displays three car categories at the top: **Užitkové** (Utility), **Osobní automobily** (Personal cars), and **Sportovní vozy** (Sports cars). Below these are three car images: a red van, a silver sedan, and a silver sports car. In the center, there are six sliders for different criteria, each with a numerical value in a box to its right:

Kritérium	Hodnota
Design	20
Zavazadlový prostor	20
Zrychlení	10
Spotřeba	15
Rychlost	10
Řízení	25

Below the sliders is a button labeled **Seřadit**. At the bottom, two car models are shown with their respective star ratings and percentages:

- Mazda cx-7**: 5 stars, 58%
- Škoda Roomster**: 5 stars, 39%

Below the ratings are images of the Mazda CX-7 (blue) and the Škoda Roomster (white).

Obrázek 11 : Ukázka možnosti manuálního řazení kontextů v kategorii osobních automobilů.

Aby bylo vůbec možné pomocí posuvníků dosáhnout požadovaného rozložení vah kontextů, ukládá si komponenta počet bodů, o které se každý posuvník posunul do pomocných proměnných,

⁵ Výjimkou je situace při vytvoření prvního kontextu kategorie. Tehdy je suma rovna nule, protože nově přidané kontexty jsou vytvářeny s nulovou vahou, aby neovlivnily nastavení vah kontextů stávajících.

kdy u posuníku, který byl nastavován ručně, je tato hodnota vynásobena dvěma. Při rozdělování bodů pak algoritmus distribuce upřednostňuje posuvníky s touto nejnižší pomocnou proměnou. Vynásobení dvěma ručně modifikovaného posuvníku zaručí, že se hodnota na tomto posuvníku už nebude chtít měnit. Pro jemné doladění je též možné donastavit hodnoty na posuvnících pomocí šipek na klávesnici.

5.4 Databázová dotazovací kritéria

Z důvodů co možná nejvyššího výkonu a nejmenšího množství zbytečně načítaných objektů pomocí objektového mapování dat na entity je v celé aplikaci základní načítání všech referovaných objektů nastaveno na LAZY, tzn. že objekty jsou namapovány jen v případě potřeby. Z důvodů množství dotazů pracujících nad těmito objekty je pak ale nutné, buď vytvořit velké množství téměř shodných funkcí lišících se pouze v těle dotazu, a nebo je potřeba sestavit ne úplně čitelnou a snadno pochopitelnou soustavu podmínek postupně vytvářejících tělo dotazu z argumentů funkce. Obdobný problém nastává nejen u webově založených aplikací, jeli nutné výsledný seznam objektů upřesnit pomocí podmínek například na maximální cenu či konkrétního výrobce.

Alternativa k tomuto postupu se nabízí v podobě vytváření dotazů pomocí Hibernate Criteria API. V kombinaci s použitím proměnného množství argumentů funkce se pak jedná o elegantní řešení umožňující pohodlné psaní dotazů. Abychom mohli používat instanci objektu Criteria, je ji nejprve nutné vytvořit pomocí factory metody třídy Session. Poté již možnosti přidávání omezení, nahrávání referujících objektů či řazení nic nebrání.

```
public Set<DiObject> findAllMyAndAncestorsObjects(DiCategory category,
    Integer resultSize, Integer pageNumber, Order order,
    String... fetchedFieldNames) {

    Criteria criteriaObject = getSessionFactory().getCurrentSession()
        .createCriteria(DiObject.class).createAlias("category", "cat")
        .add(Restrictions.ge("cat.left", category.getLeft()))
        .add(Restrictions.le("cat.right", category.getRight()))
        .setFetchMode("category", FetchMode.JOIN);

    for (String fieldName : fetchedFieldNames) {
        criteriaObject.setFetchMode(fieldName, FetchMode.JOIN);
    }

    if (order != null) {
        criteriaObject.addOrder(order);
    }
}
```

Zdrojový kód 1: Ukázka použití kritérií ve třídě DiObjectDAO.

6 Navržená vylepšení modelu a jejich implementace

Při testování aplikace vybraným vzorkem uživatelů byla odhalena nejen celá řada chyb a nedostatků ale upozornila i na vhodné možnosti rozšíření aplikace a mnoha-kontextového modelu reputace. Pro potřeby testování byla zvolena reprezentativní skupina pěti uživatelů z okruhu příbuzných a známých. Testování bylo zaměřeno na okruhy:

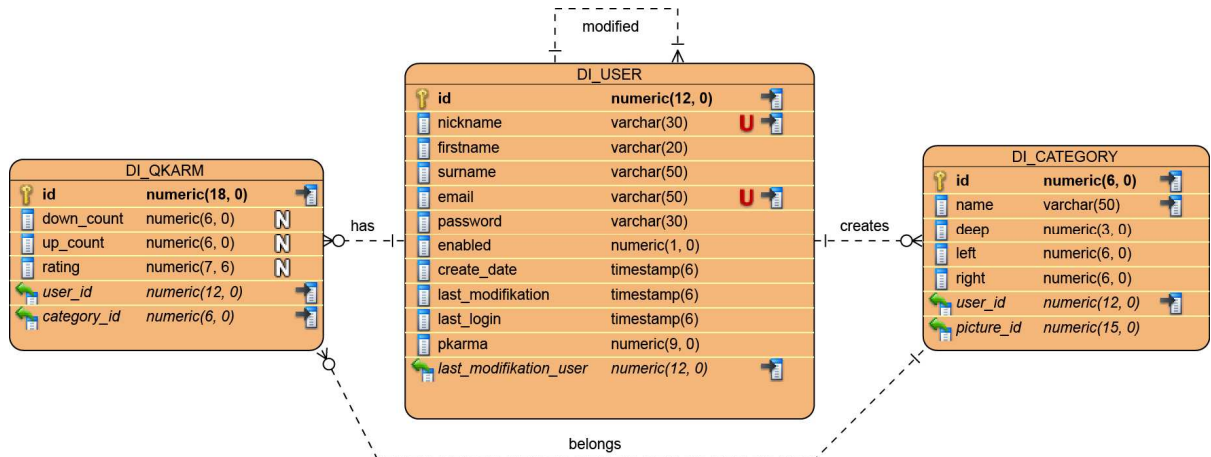
- Logika a funkčnost reputačního modelu.
- Ošetření formulářů a navigace mezi stránkami.
- Zabezpečení aplikace proti neoprávněnému přístupu.

Zatímco problémy z oblasti ošetření formulářů a navigace mezi stránkami představují díky výjimkám poměrně lehce odhalitelné a opravitelné chyby a chyby v zabezpečení jsou díky logům rovněž lehce zjistitelné. Testování funkčnosti reputačního modelu ale představuje dlouhodobější proces. Na základě testování a připomínek uživatelů bylo navrženo a implementováno několik vylepšení týkající se funkčnosti reputačního modelu. Dvě z těchto vylepšení představují následující 2 kapitoly.

6.1 Rozdělení uživatelské q-karmy podle kategorií

Algoritmus výpočtu hodnoty celkového skóre objektu používaného při řazení doposud vycházel z celkové q-karmy uživatele, která se formovala díky hodnocením recenzí ostatními uživateli napříč všemi kategoriemi. Z takového implementace pak ale vyplívá, že uživatel, který například získal vysokou q-karmu jako hardwarový specialista v rámci kategorie počítače, bude specialistou i ve všech ostatních kategoriích a jeho hlas bude mít automaticky vyšší váhu i v kategorii Bílé zboží → Narovnávače vlasů, přestože o dané problematice s největší pravděpodobností nic neví.

Řešením je tedy rozštěpit tuto obecnou q-karmu na sadu q-karem, vždy pro jednu listovou kategorii jednu q-karmu. Z implementačního hlediska představuje tato změna zásah do databázového schématu a přesun atributů součtu záporných a kladných palců do samostatné tabulky s cizími klíči na uživatele a kategorii viz obrázek 12. Na straně implementace pak bylo nutné vytvořit entitu reprezentující nově vytvořenou tabulku a potřebné servisní a DAO třídy s rozhraními.



Obrázek 12: Úprava databázového schématu pro rozdělení q-karmy podle kategorií.

6.2 Celkové skóre zohledňující množství recenzí

Původní návrh reputačního modelu popsany v kapitole 2.1.4 nepočítal s možností ovlivnění skóre objektů na základě počtu recenzí. Představme si 2 objekty ve společné kategorii. Váhy u všech hodnocení u obou objektů jsou na základní hodnotě 0.1. Stejně tak výsledky všech hodnocení budou u obou objektů stejné (q-karma recenzí zde nehraje roli, protože si jsou všechny hodnocení rovny). Objekt A bude mít ale jen dvě hodnocení oproti objektu B s hodnoceními dvaceti. Celkové skóre obou objektů podle současné vyhodnocovací funkce přesto vyjde stejně. Jelikož pro výpočet skóre objektu B posloužilo mnohem více dat, představuje pro nás tento výsledek větší jistotu, že je dané skóre správné a proto by se tato skutečnost měla i do celkového skóre promítnout.

V tomto případě je pro nás řešení využití kompenzační funkce zohledňující počet hodnocení s globálně upravenými parametry uvedené v kapitole 2.1.4. S přihlédnutím ke změnám navrženými v předchozí kapitole bude mít rovnice pro výpočet celkového skóre objektu následující podobu:

$$R = \min \left(\max \left(\frac{\sum_{i=0}^n [(r_i) * (qK_i^{uk} * pK_i^r * qK_i^r)]}{\sum_{i=0}^n (qK_i^{uk} * pK_i^r * qK_i^r)} \right) - a - \min \left(\max \left(\frac{(n-f)}{c}, 0 \right), 1 \right) * 2 * a, 0 \right), 1 \right) \quad (16)$$

kde:

- n je množství recenzí započítaných do celkového hodnocení (započítány jsou pouze ty recenze, jež se nachází v časovém intervalu kategorie, do které objekt patří),
- r_i je hodnocení recenze i ,

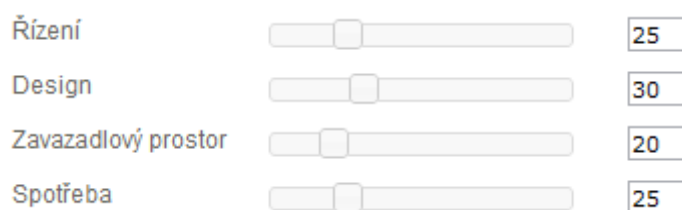
- qK_i^{uk} je hodnota q-karma uživatele recenze i pro kategorii, do níž objekt spadá, v době vytvoření recenze,
- pK_i^r je p-karma recenze i - funkce zohledňující množství palců (kladných i záporných),
- qK_i^r je q-karma recenze i ,
- a je kompenzační konstanta globálně nastavena pro všechny kategorie na hodnotu 0.1,
- f je kompenzační konstanta minimálního počtu recenzí globálně nastavena pro všechny kategorie na hodnotu 5,
- c je kompenzační konstanta maximálního počtu recenzí globálně nastavena pro všechny kategorie na hodnotu 30,
- \min je funkce vracející menší hodnotu ze dvou čísel a
- \max je funkce vracející větší hodnotu ze dvou čísel.

7 Případová studie

Tato kapitola prezentuje funkčnost navrženého mnoha-kontextového reputačního modelu a ukazuje vliv vah q-karmy uživatele, q-karmy recenze, p-karmy recenze a kompenzační funkce zohledňující počet recenzí na celkové skóre objektu.

7.1 Objekty, uživatelé a recenze

Pro potřeby prezentace s modelem byla vybrána kategorie *Osobní automobily* s kontexty: *Řízení*, *Design*, *Zavazadlový prostor* a *Spotřeba* s rozložením vah podle obrázku 13. Kategorie obsahuje čtyři objekty bez recenzí - *Dacia logan 2*, *Škoda Roomster*, *Mazda CX-7* a *Škoda Fabia*. Do hodnocení budou zasahovat 3 skupiny hodnotitelů - *experti* s maximální q-karmou, *nováčky* s neutrální q-karmou a *amatéři* s maximální q-karmou negativní.



Obrázek 13: Základní rozložení vah kontextů v kategorii *Osobní automobily*.



Obrázek 14: Objekty kategorie.

Ke každému objektu vytvořil vždy jeden zástupce ze skupiny hodnotitelů jednu svoji recenzi. Hodnoty kontextů, karem a hodnocení recenzí ovlivňujících skóre shrnují tabulky 17 a 18.

7.2 Vliv Q-karmy uživatele na skóre

Q-karma uživatele reprezentuje způsob, jak může ovlivnit zkušenost uživatele významnost recenze a tím její podíl na výpočtu celkového skóre. Ze součtu vah karem recenzí v tabulkách 17 a 18 je vidět,

že recenze expertního hodnotitele bude mít při neutrálních ostatních karmách oproti nováčkovy desetkrát vyšší váhu.

	Recenze Dacia Logan			Recenze Škoda Roomster		
	Expert	Amatér	Nováček	Expert	Amatér	Nováček
Řízení	★★★★★	★★★★★	★★★★★	★★★★★	★★★★★	★★★★★
Design	★★★★★	★★★★★	★★★★★	★★★★★	★★★★★	★★★★★
Zavaz. prostor	★★★★★	★★★★★	★★★★★	★★★★★	★★★★★	★★★★★
Spotřeba	★★★★★	★★★★★	★★★★★	★★★★★	★★★★★	★★★★★
Hodnocení	34%	5%	44%	49%	78%	47%
Q-karma uživatele	1	0,01	0,1	1	0,01	0,1
Q-karma recenze	0,1	0,1	0,1	0,1	0,1	0,1
P-karma recenze	0,1	0,1	0,1	0,1	0,1	0,1
Výsledná váha	0.01	0.0001	0.001	0.01	0.0001	0.001

Tabulka 17: Uživatelské recenze a vztahované karmy.

	Recenze Mazda CX-7			Recenze Škoda Fabia		
	Expert	Amatér	Nováček	Expert	Amatér	Nováček
Řízení	★★★★★	★★★★★	★★★★★	★★★★★	★★★★★	★★★★★
Design	★★★★★	★★★★★	★★★★★	★★★★★	★★★★★	★★★★★
Zavaz. prostor	★★★★★	★★★★★	★★★★★	★★★★★	★★★★★	★★★★★
Spotřeba	★★★★★	★★★★★	★★★★★	★★★★★	★★★★★	★★★★★
Hodnocení	66%	91%	68%	40%	80%	60%
Q-karma uživatele	1	0,01	0,1	1	0,01	0,1
Q-karma recenze	0,1	0,1	0,1	0,1	0,1	0,1
P-karma recenze	0,1	0,1	0,1	0,1	0,1	0,1
Výsledná váha	0.01	0.0001	0.001	0.01	0.0001	0.001

Tabulka 18: Uživatelské recenze a vztahované karmy.

Z výsledků celkových skóre objektů v tabulce 19 je vidět, že pokud se hodnocení nováčka zásadně neliší od hodnocení experta, bude celkové skóre prakticky kopírovat hodnotu expertní recenze. Naopak hodnocení amatéra se téměř neprojevovalo, až už bylo jakékoliv.

Automobil	Celkové skóre	Skóre po kompenzacích ⁶	Pořadí
Mazda CX - 7	66%	56%	1
Škoda Roomster	49%	39%	2
Škoda Fabia	42%	32%	3
Dacia Logan	35%	25%	4

Tabulka 19: Celkové pořadí objektů a jejich skóre při účasti uživatelské q-karmy.

7.3 Vliv P-karmy recenze na skóre

P-karma recenze reprezentuje způsob, jak může ovlivnit mnoha palci ohodnocená (a zřejmě i hodně diskutovaná mezi uživateli) recenze svoji významnost a tím její podíl na výpočtu celkového skóre. Tabulka 20 ukazuje změnu p-karmy u vybraných recenzí a objektů a tabulka 21 vliv těchto změn na celkové skóre a pořadí objektů.

	Dacia Logan			Škoda Fabia		
	Expert	Amatér	Nováček	Expert	Amatér	Nováček
Q-karma uživatele	1	0,01	0,1	1	0,01	0,1
Q-karma recenze	0,1	0,1	0,1	0,1	0,1	0,1
P-karma recenze	0,1	1	0,1	0,1	0,1	1
Výsledná váha	0.01	0.0001	0.001	0.01	0.0001	0.001

Tabulka 20: Recenze s mnoha palci. Zeleným pozadím je zvýrazněn kladný rozdíl oproti tabulkám 17 a 18.

⁶ Jelikož žádný z objektů nedosáhl ani na minimální počet recenzí, kompenzační funkce odebrala ze všech celkových skóre hodnotu 0.1.

Automobil	Celkové skóre	Skóre po kompenzaci	Pořadí
Mazda CX - 7	66%	56%	1
Škoda Fabia	50%	40%	2
Škoda Roomster	49%	39%	3
Dacia Logan	32%	22%	4

Tabulka 21: Celkové pořadí objektů a jejich skóre při účasti navíc oproti tabulce 20 i p-karmy recenze.

7.4 Vliv Q-karmy recenze na skóre

Q-karma recenze reprezentuje způsob, jak může ovlivnit pravdivost recenze (stanovená ostatními uživateli) svoji významnost a tím její podíl na výpočtu celkového skóre. Tabulka 22 ukazuje změnu q-karmy u vybraných recenzí a objektů. Tabulka 23 pak vliv těchto změn na celkové skóre a pořadí objektů.

	Mazda CX-7			Škoda Fabia		
	Expert	Amatér	Nováček	Expert	Amatér	Nováček
Q-karma uživatele	1	0,01	0,1	1	0,01	0,1
Q-karma recenze	0,01	0,1	0,1	0,1	0,1	1
P-karma recenze	0,194	0,1	0,1	0,1	0,1	1
Výsledná váha	0.0194	0.0001	0.001	0.01	0.0001	0.1

Tabulka 22: Recenze s q-karmou. Červeným pozadím je v tabulce zvýrazněn negativní rozdíl oproti tabulce 20.

Automobil	Celkové skóre	Skóre po kompenzaci	Pořadí
Mazda CX - 7	67%	57%	1
Škoda Fabia	58%	48%	2
Škoda Roomster	49%	39%	3
Dacia Logan	32%	22%	4

Tabulka 23: Celkové pořadí objektů a jejich skór navíc oproti tabulce 21 i při účasti q-karmy recenze.

Přestože váha uživatele bude mít zřejmě nejvýznamnější roli při počítání celkového skóre (minimálně určitou dobu po vytvoření recenze), neboť je první vahou, která bude skóre ovlivňovat a je s časem

neměnná. Z výsledné tabulky 23 a 22 je vidět, že i přesto může být špatné expertní tvrzení v průběhu času zvráceno.

7.5 Srovnání vlastního reputačního modelu s existujícími

Pro srovnání vlastního reputačního modelu (vystupujícím pod jménem Raputace.cz) byly vybrány dva existující modely: mnoha-kontextový reputační model *Fajnšmekr* [14], jakožto nejlepší ze zkoumaných mnoha-kontextových modelů a reputační model serveru *Heureka!* [29], který představuje přímou konkurenci naší aplikace. Tabulky 24, 25 a 26 shrnují výsledky porovnání.

Server	Variabilní	Konstantní	Vyhledávání	Složené skóre	Různá váha
Fajnšmekr	x(4)		x	x	x
Heureka!					
Reputace.cz	x(>1)		x	x	x

Table 24: Zjištěné využití kontextů.

sever	Příspěvky			Registrace	Průměr	
	Stárnutí	Karma	Množství	Povinnost	Jednoduchý	Vážený
Fajnšmekr	x	x	?	x		x
Heureka!			x			x
Reputace.cz	x	x	x	x		x

Table 25: Zjištěné základní reputační modely.

Server	Reviews	Ratings	This-or-That Voting	Points	Report abuse	Vote to Promote
Fajnšmekr	x					
Heureka!	x		x	x		x
Reputace.cz	x		x			

Table 26: Zjištěné základní reputační modely.

8 Možnosti rozšíření modelu

V této kapitole jsou popsány další možnosti rozšíření mnoha-kontextového reputačního modelu či aplikace. Společně s možnými rozšířeními jsou zde zmíněny i hlavní otevřené problémy, které vyplynuly ze stávajícího konceptu a jež se nepodařilo vyřešit či jejich změna by z konceptuálního hlediska představovala příliš velký zásah do logiky či architektury aplikace. Společně s popisem bude vždy nastíněno i možné řešení.

8.1 Rekalkule hodnot recenzí při změně váhy kontextů

Při vytváření recenzí je výsledek recenze spočítán z přednastavených vah kontextů pro danou kategorii a pro potřeby urychlení výpočtu při řazení objektů je výsledek uložen společně s recenzí. Při změně přednastavených vah či změně počtu kontextů se ale výsledky recenzí vytvořených před změnou stávají neaktuálními

Zatímco v případě změn vah kontextů by přepočítání výsledků všech recenzí pro všechny objekty v dané kategorii dávalo ještě smysl, přestože by se nejednalo z časového hlediska o zcela triviální operaci, změna počtu kontextů by již ale mohla hodnocení znevážit. V případě odebrání kontextu by se hodnota kontextu rovnoměrně rozdělila mezi stávající kontexty, kde váhou rozdělení by byly původních hodnoty ostatních kontextů (v případě, že by se rozdělovalo 20% bodů, kontext, jež držel původně 40% by vzrostl o $0,4 * (100 / 80) * 20$ bodů, kde zbytek po celočíselném dělení by obdržel kontext s největší původní hodnotou).

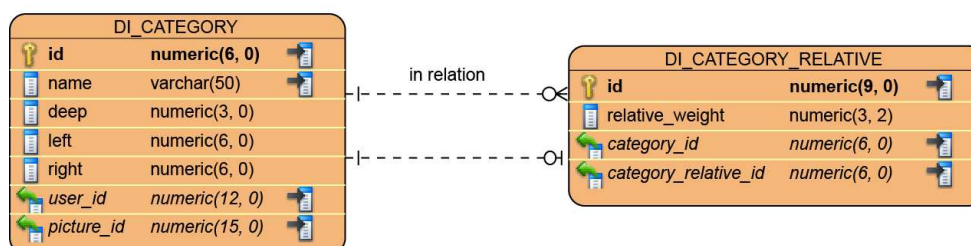
V případě přidání kontextu by se stávající hodnoty kontextů nezměnily a nový kontext by zůstal na nule, neboť se předpokládá, že při vytváření kategorie byly všechny důležité kontexty již vytvořeny a další, později přidané, nebudou tak významné. V každém případě zůstává možnost po změně počtu kontextů upravit váhy pomocí posuvníků ručně.

8.2 Vazby mezi kategoriemi

Rozdělení společné q-karmy na sadu q-karem pro danou kategorii je zajisté posunem vpřed, nicméně prostor pro další vylepšení tu stále zůstává. Některé kategorie toho totiž mají společného hodně a přeci jenom člověk, který je odborníkem přes digitální zrcadlovky, bude s velkou pravděpodobností zkušenější než většina uživatelů i v kategorii digitálních kompakťů. Nabízí se zde tedy prostor pro provázání q-karem příbuzných kategorií a jejich vzájemném ovlivňování v podobě váhy,

reprezentující míru příbuznosti. Úprava schématu by spočívala ve vytvoření nové entity *příbuzní kategorie* a jejím provázáním se stávající tabulkou *kategorie* viz obrázek 15.

Přestože palec představuje celočíselnou hodnotu (0 pro záporný, 1 pro kladný), algoritmus výpočtu q-karmy pracuje s rozdílem sum obou typů. Funkce, jež daný výpočet přebírá, může mít ale jako argument i reálnou hodnotu - *palcováhu*, je tedy tato úprava možná a přidání 0.45 palce na základě příbuznosti z 45% k sumě kladných palců nepředstavuje žádný problém.



Obrázek 15: Návrh úpravy schématu pro možnost stanovení příbuzných kategorie.

8.3 Uchovávání vlastního váhového rozložení

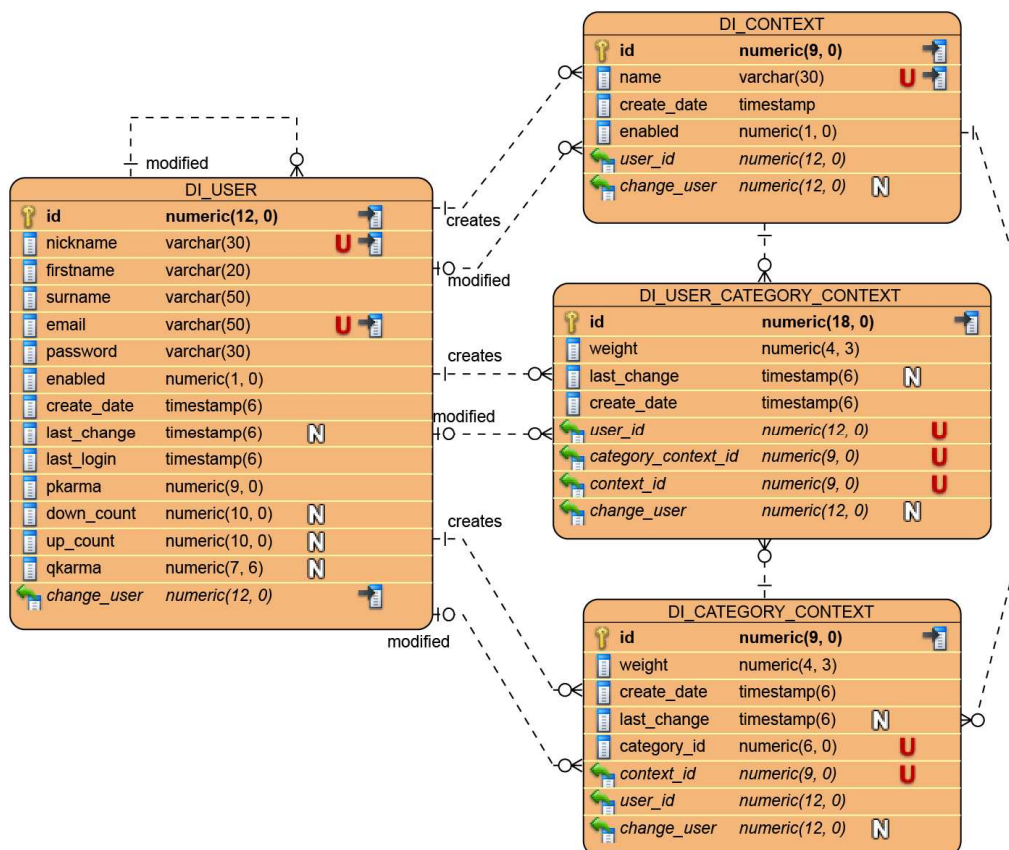
Smyslem tohoto vylepšení je umožnit uživatelům trvale uchovávat jejich vlastní váhové rozložení kontextů pro danou kategorii. Nezbytnou podmínkou pro využití této možnosti je předchozí přihlášení uživatele do systému. (Jistou alternativou by mohlo být uložení rozložení do uživatelské cookies.) Vlastní proces uložení je podrobněji popsán v detailu případu použití v tabulce 16.

ID:	1
Název:	Vlastního váhového rozložení
Vytvořeno:	Návrhář, vývojář
Popis	Uchovávání vlastního váhového rozložení
Primární aktéři	Registrovaný uživatel
Předpoklady	1. Uživatel je přihlášen do systému
Hlavní tok:	1. Uživatel nastaví váhové rozložení kontextů podle svých preferencí. 2. Zmáčkne tlačítko "Uložit vlastní rozložení".
Frekvence:	Často
Speciální požadavky:	1. Žádné

Tabulka 17 : Detail případu použití ukládání vlastního váhového kontextového rozložení.

Při dalším navštívení kategorie s vlastním nastavením vah kontextů budou přihlášenému uživateli defaultně zobrazeny objekty v pořadí dle jeho uložených preferencí. Uživateli bude také umožněno resetovat uložené nastavení pomocí tlačítka "Základní nastavení" a vrátit se tak k základnímu přednastavenému rozložení vah kontextů pro danou kategorii.

Z pohledu databázového schématu vyžaduje rozšíření přidání nové tabulky s vazbami na kontexty, uživatele a vazební tabulku kategorie a kontexty, kdy zároveň tato trojkombinace cizích klíčů slouží v nové tabulce jako unikátní klíč. Přesná podoba rozšíření je znázorněna na obrázku 17.



Obrázek 17: Návrh rozšíření databázového schématu pro možnost uchovávání vlastního kontextového rozložení.

8.4 Pokročilejší implementace modelu stárnutí

Stávající implementace aplikace využívá nejprimitivnější model stárnutí časově omezeného přepočtu objektů, jehož největší nevýhodou je nízká efektivnost. Při každém požadavku se z databáze vyberou záznamy, jež splňují podmínku pevně stanoveného časového intervalu, a nad nimi se provede výpočet. V případě, že nešlo o dotaz s vlastním nastavením vah kontextů, bude výsledek ve většině případů stejný, jako v předchozím dotazu. V případě rozsáhlé komunity uživatelů a objektů se v případě takového řešení dají očekávat problémy s vysokou zátěží databázového serveru.

Možností zlepšení současného řešení se zde nabízí celá řada. Jedním z řešení na úrovni databáze může být vytvoření materializovaného pohledu, jež výsledky předpřipraví. V době nejmenší zátěže serveru, se pak jedenkrát za den tento pohled aktualizuje. Při řazení se pak místo vlastního výpočtu pouze použije takto předpřipravené skóre objektu.

9 Závěr

Tato diplomová práce navázala na semestrální projekt, v rámci kterého byly položeny teoretické základy týkající se témat důvěry, reputace, reputačních a mnoha-kontextových reputačních modelů. Na základě studia reálných mnoha-kontextových reputačních systémů provedeném v semestrálním projektu, byl vytvořen návrh vlastního mnoha-kontextového modelu a posléze i jeho implementace zasazená do webové aplikace postavené na frameworku Spring v prostředí J2EE.

Samotná aplikace byla navržena jako obecný hodnotící systém, rozdělující hodnocené objekty podle typu hierarchicky do kategorií. Stejně tak jako umí aplikace dynamicky vytvářet kategorie, umí vytvářet i kontexty a tyto kontexty společně s jejich váhou přidávat ke kategoriím.

Při výpočtu celkových skóre objektů byly brány v potaz kromě samotného hodnocení z recenzí i váhy v podobě karem. Kromě quality karmy recenze (pravdivost recenze) a participation karmy (množství palců) recenze ovlivňuje celkové skóre i quality karma autora recenze (dělení na experty či amatéry), jež je vlastní pro každou kategorii.

Na základě studia reputačních systémů v semestrálním projektu můžeme konstatovat, že mnoho-kontextovost je spíše vzácností. V případě, že reputační systém umožňuje zaznamenávat mnoha-kontextovou podstatu objektu, nedokáže ji většinou vhodně využít a navíc mnohdy podává rozporuplné výsledky. Ukázkou naprosto nevyužitého potenciálu je, že ze dvanácti sledovaných systémů pouze jeden umí podle těchto kontextů také řadit a vyhledávat.

Pro potřebu řazení a vyhledávání mezi objekty byla v aplikaci vytvořena speciální komponenta - *vícenásobný posuvník*, umožňující měnit váhové rozložení kontextů a tím objekty řadit, a to dokonce i pro více kontextů naráz.

Co se týče možnosti dalšího rozšíření reputačního modelu, za zmínku by rozhodně stála možnost ukládání vlastního váhového rozložení kontextů kategorie, na jehož základě by se objekty přirozeně řadily, či stanovovat příbuzné kategorie včetně míry příbuznosti, jejichž smyslem by bylo určování pravděpodobných expertů v podobných kategoriích.

Přestože nějaká forma reputace je přítomna prakticky na jakémkoliv významnějším serveru, mnoho-kontextovost se ještě standardní součástí reputačních modelů nestala a nabízí se zde tak stále dostatek prostoru pro zlepšení a inovaci.

Literatura

- [1] FARMER, R., GLASS Bryce. : *Building Web Reputation Systems*. 1. edition London: O'Reilly Media / Yahoo Press, 2010. 336 s. ISBN 978-1-4493-8256-8
- [2] AXELROD, Robert. *The Evolution of Cooperation*. New York : Basic Books, 1984. 256 s. ISBN 0-465-02122-2.
- [3] The Market for "Lemons" : Quality Uncertainty and the Market Mechanis. *The Quarterly Journal of Economics*. 1970, 84, s. 488-500. Dostupný také z WWW: <<http://163.117.2.172/microii-phd/G%20Akerlof.pdf>>.
- [4] ABDUL-RAHMAN, Alfarez, HAILES, Stephen. Supporting trust in virtual communities. *In 33rd Hawaii International Conference on System Sciences*, 6:6007, 2000.
- [5] WINKLER, Jiří; PETRUSEK, Miloslav. *Velký sociologický slovník*. Praha : Karolinum, 1997. 598 s. ISBN 80-7184-164-1.
- [6] MUI, Lik; MOHTASHEMI, Mojdeh; HALBERSTADT, Ari. A Computational Model of Trust and Reputation for E-businesses. *In Proceedings of the 35th Annual Hawaii International Conference on System Sciences : HICSS'02*, 2002. 188 s. ISBN 0-7695-1435-9.
- [7] VOSS, Marco; BRÜCKNER, Lars; SCHLOSSER, Andreas. *Positional Reputation Systems* [online]. Darmstadt : Department of Computer Science Darmstadt University of Technology, 200? [cit. 2011-01-08]. Dostupné z WWW: <http://www.ito.tu-darmstadt.de/publs/pdf/VossEtAl_Positional_Reputation_Systems.pdf>.
- [8] SABATER, Jordi; SIERRA, Carles Regret : reputation in gregarious societies. *In Proceedings of the fifth international conference on Autonomous agents*, 2001. s. 194-195.
- [9] MUI, Lik; *Computational Model sof Trust and Reputation: Agents, Evolutionary Games, and Social Networkds*. PhD thesis, Massachusetts Institute of Technology, 2003.
- [10] *EBay* [online]. 1995 [cit. 2010-12-26]. Dostupné z WWW: <<http://www.ebay.com/>>.
- [11] *The Internet Movie Database* [online]. 1990 [cit. 2010-12-26]. Dostupné z WWW: <<http://www.imdb.com/>>.
- [12] *Best Buy* [online]. 2003 [cit. 2010-12-28]. Dostupné z WWW: <<http://www.bestbuy.com>>.
- [13] *Nike* [online]. 2003 [cit. 2010-12-28]. Dostupné z WWW: <<http://www.nike.com>>.
- [14] *Fajnšmekr* [online]. 2009 [cit. 2010-12-28]. Dostupné z WWW: <<http://www.fajnsmekr.cz/>>.

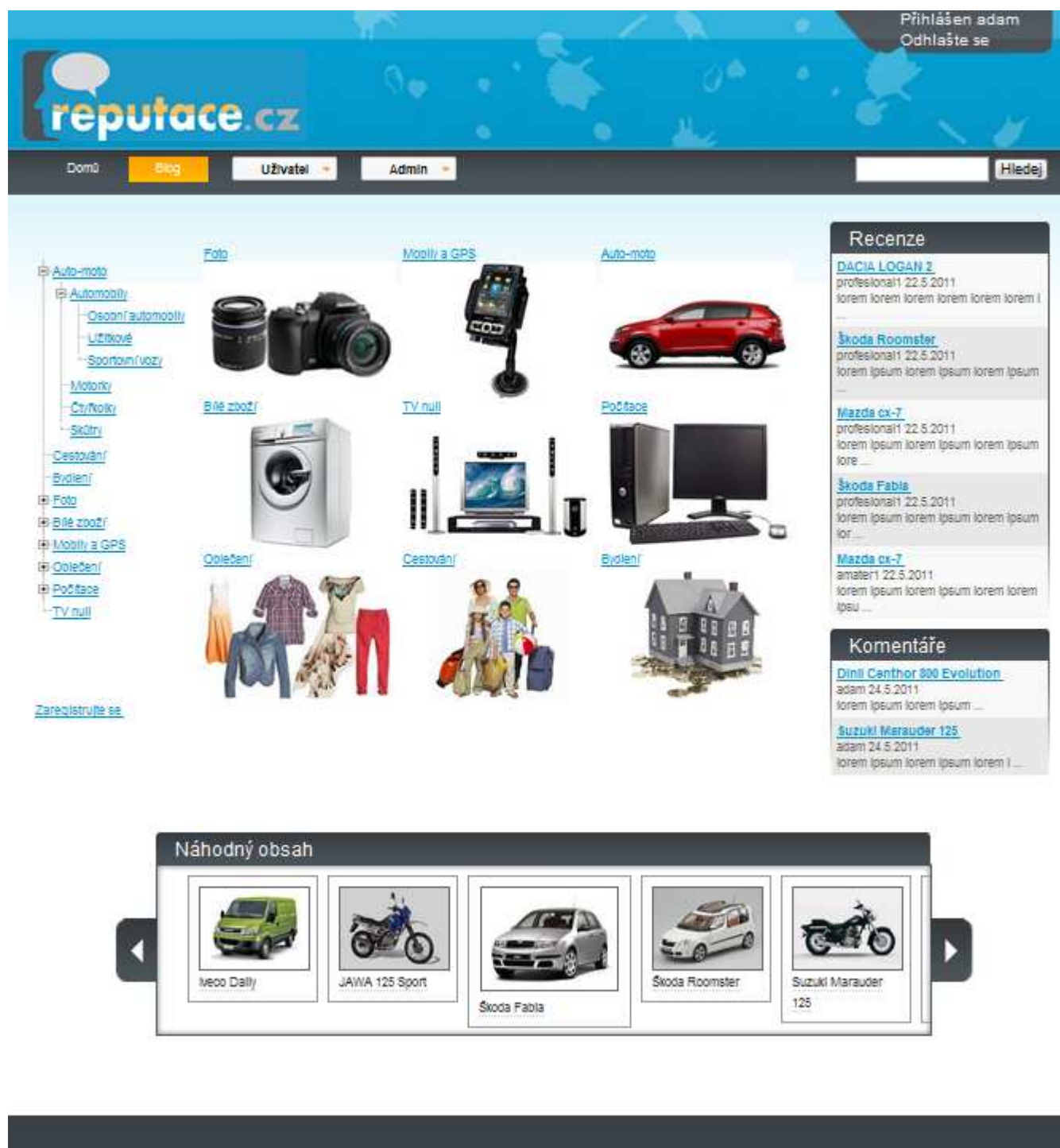
- [15] *NetTravel* [online]. 2003 [cit. 2010-12-28]. Dostupné z WWW: <<http://www.nettravel.cz/>>.
- [16] *Dealextreme* [online]. 2007 [cit. 2010-12-28]. Dostupné z WWW: <<http://http://www.dealextreme.com/>>.
- [17] *TEST FREAKS* [online]. 2007 [cit. 2010-12-29]. Dostupné z WWW: <<http://www.testfreaks.com/>>.
- [18] *Yahoo!* [online]. 2000 [cit. 2010-12-29]. Dostupné z WWW: <<http://www.yahoo.com/>>.
- [19] *Crest* [online]. 2002 [cit. 2010-12-30]. Dostupné z WWW: <<http://www.crest.cz/>>.
- [20] *Review centre* [online]. 2003 [cit. 2010-12-30]. Dostupné z WWW: <<http://www.reviewcentre.com>>.
- [21] *Restaurantica* [online]. 2008 [cit. 2011-1-1]. Dostupné z WWW: <<http://www.restaurantica.com>>.
- [22] *Hotel Thailand* [online]. 2005 [cit. 2011-1-1]. Dostupné z WWW: <<http://www.hotelthailand.com>>.
- [23] *What car?* [online]. 2004 [cit. 2011-1-1]. Dostupné z WWW: <<http://www.whatcar.com>>.
- [24] *Web Hosting Geeks* [online]. 2004 [cit. 2011-1-6]. Dostupné z WWW: <<http://webhostinggeeks.com>>.
- [25] *App Store* [online]. 2006 [cit. 2011-1-8]. Dostupné z WWW: <<http://www.apple.com/iphone/features/app-store.html>>.
- [26] MACHACEK, Jan, et al. *Pro Spring 2.5*. 1 edition. [s.l.] : Apress, 2008. 920 s. ISBN 978-1-59059-921-1.
- [27] LINWOOD, Jeff; MINTER, Dave. *Beginning Hibernate*. second edition. New York : Apress, 2008. 400 s. ISBN 978-1-4302-2850-9.
- [28] *BOOKING* [online]. 1996 [cit. 2011-5-20]. Dostupné z WWW: <<http://www.booking.com>>.
- [29] *Heureka!* [online]. 2000 [cit. 2011-5-21]. Dostupné z WWW: <<http://www.heureka.cz>>.

Seznam příloh

A Galerie aplikace

B Datový nosič CD

A Galerie aplikace



Obrázek 18: Titulní stránka webové aplikace.